Ищите ваши файлы правильно с. 14

LXFDVD

Ubuntu 9.10
+ OpenSUSE 11.2
+ Mandriva 2010

ПЛЮС: MythTV, Seamonkey, Tiny Core и прочие!

LINUX FORMAT

Главное в мире Linux

Январь 2010 № 1 (126/127)

28 страниц учебников на любой вкус!

Пусть ваш Linux выглядит круто!

Приведите свой рабочий стол в соответствие со стандартами 2010 года



Plan 9
Так мог бы выглядеть Unix, если бы его придумали на ...дцать лет позже с. 34

Сетевая безопасность
Перехват данных на службе у Добра с. 44



«Те, кому не нравилась GPLv2, вдруг решат, что лучше и не придумаешь»
Брэдли Кун в защиту Affero с. 30

Плюс!
» Подводные лодки
» Трюки и хитрости
» Grub и мультизагрузка

» Google Chrome
Попробуйте сами и решите, радоваться или... с. 56

» OpenOffice.Org
Обрабатывайте опросники и тесты одним макросом с. 72

» Privoxy
Враг не пройдет: его остановит прокси-сервер с. 84

Подписной индекс в каталоге агентства «Роспечать» 20882
Подписной индекс в каталоге «Пресса России» 87974



Решите проблемы **лицензирования ПО** с помощью профессионалов!

Операционная система GNU/Linux и свободное программное обеспечение помогут вам с минимальными затратами решить проблему лицензирования программного обеспечения, повысить безопасность и надежность вашей компьютерной сети.

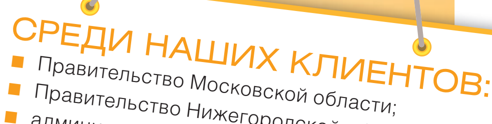
Компания ГНУ/Линуксцентр предлагает вам внедрение ОС GNU/Linux и свободного программного обеспечения, реализацию и техническую поддержку сложных технических решений на базе свободного ПО, обучение ваших сотрудников — как пользователей, так и технических специалистов.

С НАШЕЙ ПОМОЩЬЮ ВЫ СМОЖЕТЕ:

- оптимизировать затраты на лицензирование ПО за счет максимально возможного использования свободного ПО;
- существенно сократить время системных администраторов, затрачиваемое на устранение последствий деятельности вирусов и сбоев в программном обеспечении.

ТИПОВЫЕ ПРОЕКТЫ:

- миграция рабочих станций и серверов с Microsoft Windows на GNU/Linux;
- установка 1С на серверах и рабочих станциях под управлением GNU/Linux;
- миграция с Microsoft Windows Active Directory на Mandriva Directory Server;
- миграция с Microsoft Exchange на Zimbra;
- внедрение интернет-телефонии на базе Asterisk;
- внедрение свободной CRM-системы SugarCRM;
- создание кластеров высокой доступности;
- реализация терминальных решений;
- создание порталов любой сложности на базе свободных CMS-систем — Joomla, Drupal, Plone;
- внедрение защищенных систем на основе сертифицированного ФСТЭК ПО.



СРЕДИ НАШИХ КЛИЕНТОВ:

- Правительство Московской области;
- Правительство Нижегородской области;
- администрация Черниговского района Приморского края;
- Министерство финансов республики Саха (Якутия);
- Владивостокский государственный университет экономики и сервиса;
- группа компаний «ИМАГ»;
- компания «Азбука мебели»;
- компания «Бестли — выставочные материалы»

и другие организации различного профиля.

Наш опыт внедрения свободного программного обеспечения в компаниях различного профиля поможет выбрать оптимальное сочетание свободного и коммерческого программного обеспечения, подходящее именно для вашей организации, а также поможет избежать техничеких и организационных проблем при внедрении свободного ПО.



Департамент внедрений компании ГНУ/Линуксцентр
Телефон в Москве: (499) 271-49-54,
в Санкт-Петербурге: (812) 309-06-86





Что мы делаем

» Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.

» Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.

» Мы выпускам весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.

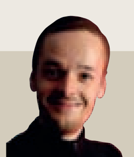
» Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.

Кто мы

В этом месяце мы приложили все усилия, чтобы Linux выглядел потрясающе. А потом решили спросить команду **LXF**: а какое ваше любимое украшение для рабочего стола?



Грэм Моррисон
Иногда у меня случаются приливы оптимизма, но моя 3D RSS-читалка быстро охлаждает пыл.



Майк Сондерс
Танцующий Столлмен – он закрывает окна несвободных приложений и добавляет GNU ко всему, что видит.



Нейл Ботвик
Ну, один раз я отредактировал **.twmrc** и поменял цвет панелей на красный. Это было круто!



Эфраин Эрнандес-Мендоса
У меня есть плазмоид, постоянно крутящий лучшие голы сборной Мексики. Да, *авто* – тоже.



Эндрю Грегори
Пони, улыбки и картинки Робби Вильямса, которые нарисовала там дочка. Прямо по монитору!



Энди Ченнел
Иконка, отображающая загрузку системы в виде индикатора здоровья из Doom. Такая есть, честно!



Дэвид Картрайт
Рабочий стол? Это такая штука вокруг сессии *Emacs*? Мне надо спросить совета у доктора M-x.



Энди Хадсон
3D-столы – вчерашний день. У меня 4D, и он перемещается во времени. Вот потому я и опоздал на работу.



Ник Вейч
Я! Большой, красивый я. Что вы говорите – я не Ди Каприо? Да ну вас всех, ей-Богу.



Сюзан Линтон
Тени – такие черные и глубокие, что свет не может покинуть поверхность окон. Поговорим о горизонте событий?



Шашанк Шарма
Украшения?.. Я что, женщина?! Верхом эстетики для сурового админа должна быть решетка в **bash #**.



Маянк Шарма
Как-то раз я рассыпал по столу пакетик Skittles и вот тут нашел одну... Эй, а про что вопрос?



На вкус и цвет

Давным-давно, когда я был простым системным администратором в одной организации, меня просто поражали рабочие столы некоторых сотрудниц. Мяукающие котята, летающие шарики, собачки, неизвестные мне молодые люди и что-то из Амели к концу рабочего дня сливались в бесконечный калейдоскоп, и я с трудом понимал, как в такой обстановке можно работать (сказать по правде, там никто особо и не работал, но нашего с вами разговора это не касается). С появлением Windows XP ситуация еще более усугубилась, и первой настройкой, которую я выполнял после ее установки, всегда становилось включение классического вида всюду, где только можно.

Когда мы заканчивали работу над этим номером журнала, мне вдруг написал старый знакомый – давний, но не безнадежный пользователь Windows, человек творческий профессии. Как выяснилось, он переводит на Linux свой мини-офис и выбирает себе дистрибутив. Из опробованных им вариантов я слышал только про OpenGEU, но (честно сказать) никогда не думал, что кто-то поставит его на рабочий ПК. Свои предпочтения знакомый сформулировал так: «По функциям они все одинаковы: надо, чтобы было красиво!»

Шаттлворт, безусловно, прав: красота – это функция. Проблема лишь в том, что понятие красивого очень субъективное. К тому-то я и веду: взяв за основу Linux и вооружившись этим номером **LXF**, вы сможете превратить свой рабочий стол во что угодно: клон Windows 95 или 7, OS X, Amiga или нечто, не похожее ни на что другое. Творите – а я, пожалуй, пойду отредактирую свой **~/.config/qtcurvestylerc**...

Валентин Синицын, главный редактор
» info@linuxformat.ru

Как с нами связаться

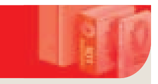
Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Web-сайт: www.linuxformat.ru
› Адрес редакции: Россия, Санкт-Петербург, Лиговский пр., 50, корп. 15
› Телефон редакции: (812) 309-06-86. Дополнительная информация на с. 112

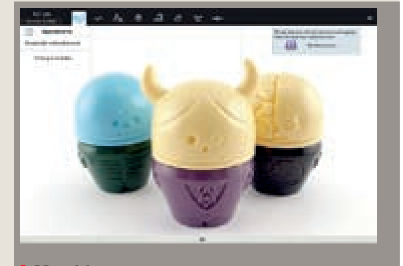


Содержание

Весь номер – прямо как на ладони: приятного чтения!

Обзоры





› **Mandriva хорошо подготовилась к этому релизу, и готова работать даже на нетбуке с крохотным экраном**



Стилист для Linux

Сразимся с Windows 7 и Mac OS X за право называться самой красивой настольной ОС с. 24

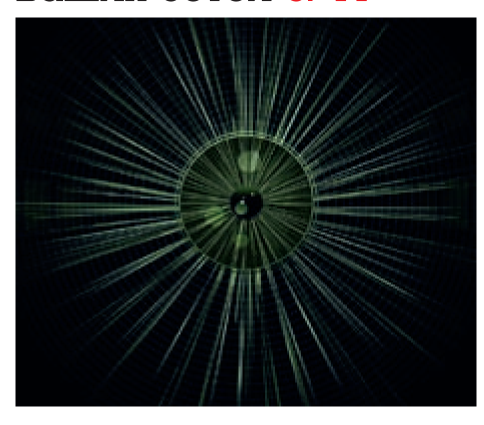
Что за штука...

Maemo

И почему Боб Мосс не разделяет опасений Евгения Крестникова? с. 48



Wireshark: на страже ваших сетей с. 44



Люди говорят



«Я-то хотел сделать иначе, но и такой вариант меня устраивает.»

Брэдли Кун умеет уважать чужое мнение с. 30

Три главных дистрибутива

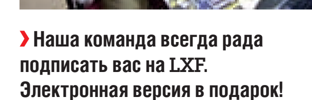
Прочитали обзоры — попробуйте сами!

LXF DVD126/127

» **Ubuntu 9.10 LXF** Наша традиционная сборка с дополнительным ПО
» **Mandriva 2010 Free** Рабочий стол, по которому скучали многие
» **OpenSUSE 11.2** Попробуйте и поставьте на жесткий диск
...плюс игры, приложения и средства разработки! с. 100

Ищите в этом номере...

Постоянные рубрики



› **Наша команда всегда рада подписать вас на LXF. Электронная версия в подарок!**

Учебники





› **О боже, сколько градиентов, отражений, теней — и все в одном доке!**

2 месяца

Лучший свободный проект России

129 участников

честная борьба

500 000 рублей

6 лауреатов

Итоги

2 победителя

В LXF122 мы объявили конкурс «Лучший свободный проект России», организованный нашим журналом с целью выявить, какие свободные проекты разрабатываются на территории Российской Федерации и стран СНГ, и наградить наиболее успешные из них.

В этом нам любезно согласились помочь корпорация Intel, компании PingWin Software, «ГНУ/Линуксцентр» и WiMaxStore, а официальную поддержку конкурсу оказало Министерство связи и массовых коммуникаций Российской Федерации (письмо № АС-П13-5822 от 20 октября 2009 года). Информационную поддержку осуществляли проект Fireforge (www.fireforge.net), электронный журнал «Open Source» (osa.samag.ru) и портал Linux.Ru.

Конкурс проходил два месяца — с 15 сентября по 15 ноября 2009 года. За это время на его официальном сайте, www.linuxformat.ru/foss-contest, было зарегистрировано 129 свободных проектов, разрабатывающихся на территории нашей страны и ее ближайших соседей. Из них Экспертное жюри в составе Александра Жмурко (PingWin Software), Александра Казанцева (EduMandriva.Ru), Вячеслава Калошина (PingWin Software), Вадима Сухомлинова (Intel) и Виктора Федосеева (Linux.Ru) отобрало по три лучших проекта в индивидуальном (1–2 основных разработчика) и групповом зачетах. Итак, мы с радостью представляем вам победителей конкурса «Лучший свободный проект России»!

НОМИНАЦИЯ «ЛУЧШИЙ ИНДИВИДУАЛЬНЫЙ ПРОЕКТ»

I место nginx — приз **100 000** рублей
Отказоустойчивый HTTP-сервер, HTTP-прокси сервер и почтовый прокси-сервер для ресурсов с высоким трафиком
http://sysoev.ru/nginx/

II место madwimax — приз **80 000** рублей
Созданный методом инженерного анализа Linux-драйвер для устройств доступа к сетям Mobile WiMAX (802.16e), выполненных на основе чипа Samsung CMC-730
http://code.google.com/p/madwimax/

III место Open Ticket Printer Drivers/OTPD — приз **70 000** рублей
Открытые кросс-платформенные драйверы принтеров чеков и этикеток
http://develoPer.berlios.de/projects/otpd/

Помимо официальной тройки победителей, мы хотели бы особо отметить еще несколько проектов: учетную систему для разработки бизнес-приложений **Ананас**, учебную систему программирования **Кумир** (читайте про нее на стр. 109), кросс-платформенный клиент сетей обмена мгновенными сообщениями **qutIM**, интерактивный инструмент для пост-обработки сканированных страниц **Scan Tailor** и web-интерфейс для системы резервного копирования **Bacula** — **Webacula**. Они просто «дышат в спину» лидерам и, мы считаем, в следующий раз им обязательно должно повезти!

Не обошлось и без интриги После объявления предварительных результатов конкурса, автор проекта 7-Zip (www.7-zip.org), единогласно признанного Экспертным жюри лучшим в номинации «Индивидуальный проект», отказался от своей премии. Поэтому, в соответствии с пунктом 4.6 официального Регламента, был проведен дополнительный отборочный тур. В результате трудной (но честной) борьбы в тройку лидеров попал **OTPD**. Пользуясь случаем, отметим, что два из шести проектов-победителей занимаются разработкой драйверов устройств — надеемся, это расширит спектр поддерживаемого Linux потребительского и специализированного оборудования и привлечет в мир свободного ПО новых людей.

НОМИНАЦИЯ «ЛУЧШИЙ ГРУППОВОЙ ПРОЕКТ»

I место CuneiForm — приз **100 000** рублей
Система оптического распознавания текстов российской компании Cognitive Technologies
http://www.cuneiform.ru

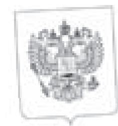
II место sK1 — приз **80 000** рублей
Редактор векторной графики, ориентированный на профессиональное применение в печатной индустрии и поддерживающий проприетарные форматы изображений
http://sk1project.org

III место Midnight Commander — приз **70 000** рублей
Двухпанельный консольный файловый менеджер
http://www.midnight-commander.org

Благодаря спонсорам первоначальный призовой фонд конкурса удалось увеличить до 600 000 рублей, что позволило наградить специальным призом «Выбор редакции Linux Format» (50 000 рублей) еще два проекта: Kalpa.Cloud и Электронный деканат (Free Dean's Office).

Мы от души поздравляем победителей конкурса, а также всех тех, кто принял в нем участие!
Думаем, что он будет не последним, проводимым Linux Format, — внимательно следите за новостями в журнале и на нашем сайте!

Лучший свободный проект России



МИНИСТЕРСТВО СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ (Минкомсвязь России)

ЗАМЕСТИТЕЛЬ МИНИСТРА

ул. Тверская, 7, Москва, 125375
Справочная: 771-81-00
по документам: 771-81-21
Факс: 771-87-18
Телетайп: 114158 MSRF RU
E-mail: office@minsvyaz.ru
Сайт: www.minsvyaz.ru

25.12.2009 № АС-П13-7413

на № ______

Генеральному директору
ГНУ/ЛИНУКСЦЕНТРА
ЗАО «Мезон.ру»

П. А. Фролову

Загородный проспект, д. 38,
лит. А, пом. 9Н
г. Санкт-Петербург 191119

ПРИВЕТСТВЕННОЕ СЛОВО ЗАМЕСТИТЕЛЯ МИНИСТРА СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ А.А. СОЛДАТОВА К УЧАСТНИКАМ КОНКУРСА «ЛУЧШИЙ СВОБОДНЫЙ ПРОЕКТ РОССИИ»

Дорогие друзья!

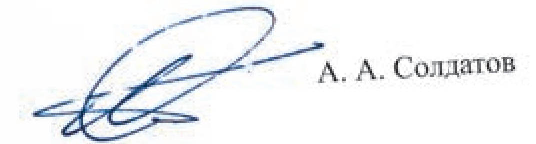
От лица Министерства связи и массовых коммуникаций Российской Федерации рад поздравить вас с успешным завершением конкурса.

В последние годы отрасль информационных технологий развивается в нашей стране усиленными темпами. Большое количество отечественных программных продуктов приобрело широкую популярность не только в России, но и по всему миру. Немаловажен и тот вклад, который вносят российские разработчики в дело развития свободного программного обеспечения.

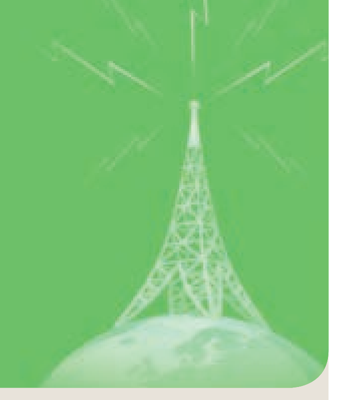
Количество, качество и многообразие работ, представленных для участия в конкурсе, позволяют убедиться не только в наличии множества отечественных специалистов в области СПО, но и в существовании широких перспектив использования свободного программного обеспечения в повседневной деятельности.

Надеюсь, что проведение подобных мероприятий станет доброй традицией для программистского сообщества, поможет поддерживать высокий интерес к информационным технологиям у граждан России, а также будет способствовать появлению новых интересных и востребованных проектов в области свободного программного обеспечения.

Заместитель Министра связи и массовых коммуникаций Российской Федерации



А. А. Солдатов



LINUX FORMAT Новости

ГЛАВНОЕ: Как будет выглядеть мобильный Linux в 2010 году?
» Войны браузеров возобновятся? » Зачем в Linux коммерческий антивирус?



ЕВГЕНИЙ КРЕСТНИКОВ
ведущий рубрики

Итоги-2009

Отгремели праздники, и перед вами – первый выпуск **LXF** в 2010 году. Поэтому часть рубрики я хочу посвятить подведению итогов прошедшего года и обсуждению перспектив наступившего. В 2009 году произошло много хорошего: совершенствовалось ядро, выходили новые версии дистрибутивов и популярных программ. Любимая многими среда KDE 4 наконец стала относительно стабильной, а Gnome плавно движется к третьей версии (уже доступен *gnome-shell*, который заменит традиционные панели). Мобильный Linux активно осваивает рынок коммуникаторов, да и компьютеры с ним уже не редкость на прилавках магазинов. В скором времени ожидается появление нетбуков с процессорами ARM.

Конечно, в прошедшем году были и не очень радостные новости, но я бы отнес их скорее к разряду курьезов. Согласитесь, разве можно всерьез говорить, например, об американских патентных войнах? Они максимум тянут на пикантную приправу к бокалу пива. Так что я еще раз поздравляю всех поклонников свободного ПО с прошедшими праздниками и перехожу к самым интересным событиям конца прошлого года.



› Вот он — обязательный Эксклюзивный Экранный Снимок «убийцы Windows 7». На самом деле, вы можете сделать такой же — да и смотреть тут особо не на что.

В зоопарк завезли пингвинов

Обсудим основные тенденции развития мобильных Linux-систем в 2010 году.

На мобильном фронте без перемен: все те же «телефонные» платформы и ОС для нетбуков и MID. Конечно, есть и «универсалы» вроде Android – поставки нетбуков с этой системой должны начаться уже в первой половине 2010 года. Android сейчас ставят повсюду, от навигаторов до игровых приставок. Среди мобильных Linux-платформ разработка Open Handset Alliance, пожалуй, перспективнее всех. И неудивительно: Google умеет продвигать продукты и сервисы. В конце года появилось (или было анонсировано) множество новых «гуглогаджетов», в основном от малоизвестных производителей. Крупные фирмы тоже не забывают об этой платформе, но планируют использовать ее в основном для коммуникаторов. Налицо тенденция снижения стоимости устройств под управлением Android, в основном за счет появления на рынке игроков второго эшелона.

> «Наилучшие перспективы у Linux — в телефонном сегменте.»

По части нетбуков, шансы продвижения у Android и других Linux-платформ я связываю с машинами на процессорах ARM. Пока это скорее мифические персонажи, которых никто не видел, но о которых много говорят. В «железе» воплощены только образцы китайской наколенной сборки да редкие изделия известных компаний, не являющихся лидерами этого рынка. В декабре по Сети поползли слухи, что нетбуки на ARM планируют выпустить ASUS и Lenovo. Представители ASUS отказались комментировать эту информацию, а сотрудники Lenovo подтвердили, что такие разработки ведутся. Готов спорить, что эти нетбуки окажутся с Android или Chrome OS. Последняя уже представлена публике – правда, пока в «негугловом» воплощении; ее исходные тексты есть на сайте проекта. В Сети найдутся и готовые образы для загрузки в *VMwave/VirtualBox*. Chromium OS – самый обычный тонкий клиент для работы с web-приложениями; однако его презентация вызвала сильный резонанс в сетевых СМИ.

По-моему, несмотря на радужные прогнозы аналитиков, нетбуки – не самая перспективная ниша для Linux. Устройства с процессорами ARM и бюджетные машины на Intel Atom – максимум, на что мы можем рассчитывать. Лидировать будет все тот же Android, а также Moblin, активно продвигаемый корпорацией Intel. Неплохие шансы у Ubuntu Netbook Remix, а Chrome OS пойдет «в народ» ближе к концу года и большой популярности не завоюет: покупатель пока не готов отказаться от локальных систем в пользу тонких клиентов.

Итак, наилучшие перспективы у Linux по-прежнему в «телефонном» сегменте, и здесь опять же лидирует Android. Кроме «гуглофонов», будет выпущено небольшое количество машин на LiMo и другом мобильном Linux. Рассчитывать на Maemo 5 напрасно; думаю, что в этом году выйдут одна-две подобные новинки, не более. К тому же Nokia никогда не заявляла, что резко откажется от Symbian S60 в пользу Linux.

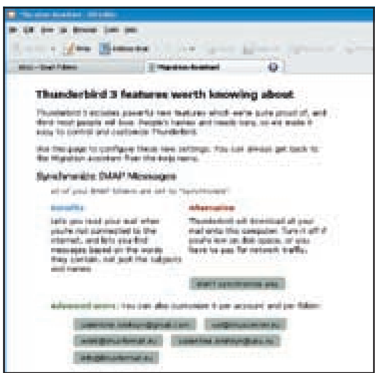
Зато интересный новогодний подарок приготовила компания Samsung, открыв в декабре доступ к анонсированной ранее Bada SDK. Похоже, корейцы всерьез решили создать свою открытую платформу на базе Linux. Других заслуживающих внимания «мобильных» событий 2009 год не принес, как не принес и сюрпризов.

Google Chrome и все-все-все

Благодаря *Mozilla Firefox*, монополярный web-мир потихоньку канул в Лету, но не стоит ли он на пороге очередной войны браузеров?

От мобильных платформ перейдем к браузерам. В декабре *Chrome* для Linux получил статус бета-версии, а в репозитории Google появился пакет *chrome-beta* (раньше там был только *chrome-unstable*). Рядовое событие, как это ни странно, вызвало сильный резонанс, в основном на тему реализованных в *Chrome* функций «большого брата». Ситуацию подогрело и опубликованное порталом Gawker интервью с исполнительным директором Google Эриком Шмидтом [Eric Schmidt]. По-моему, ничего криминального в нем нет – любой сервис-провайдер обязан передавать информацию о своих абонентах по оформленному в соответствии с законом запросу правоохранительных органов. И да, подобная информация действительно может стать основанием для повестки в суд. Вы об этом не знали? Если же говорить о том, какие данные собирает Google – что ж, устанавливая ПО или регистрируясь в сервисе, стоит внимательно прочесть текст договора (лицензионного соглашения). Но, повторяю, любой провайдер будет сотрудничать с правоохранительными органами, и Google в этом смысле вовсе не исключение. Кроме того, всегда можно воспользоваться созданным на основе *Cromium* браузером *Iron* – в нем «следящие» функции Google отключены.

Интереснее развитие интриги между Google и Mozilla. У последней в скором времени закончится контракт с поисковым гигантом, и как будут развиваться их взаимоотношения дальше, пока не понятно. Впрочем, в настоящий момент *Firefox* бурно разрабатывается и, по мнению аналитиков, остается самым популярным открытым браузером, хотя *Chrome* начинает активно его теснить. Другие продукты сильно уступают лидерам, но в завершение я бы отметил ноябрьский релиз *Opera 10.10*, в которой появилась технология Unite. Если вкратце, она позволяет превратить браузер в сервер, выполняющий приложения, написанные на JavaScript. С помощью этой разработки пользователи могут обмениваться данными через Интернет, общаться в социальной сети My Opera или организовать хостинг статичных web-страниц. Технология, безусловно, интересная, но я сильно сомневаюсь в ее популярности у рядовых пользователей.



› Ситуация на рынке web-браузеров вновь накаляется, но у Mozilla есть туз в рукаве: почтовый клиент *Thunderbird*, у которого открылось второе, а точнее — третье дыхание.

Награды «IT-Планеты 2009»

Всероссийская студенческая Олимпиада в сфере информационных технологий подошла к своему логическому завершению

В каждом из пяти федеральных округов страны, в которых прошли региональные очные туры Олимпиады, победители награждены медалями, дипломами и призами. 70 золотых медалистов будут приглашены на российский финал, который пройдет весной 2010 года в Сочи.

Всего же в образовательном проекте «IT-Планета 2009» приняли участие более 6300 студентов из 570 российских учреждений среднего и высшего профессионального образования. Каждому из них на официальном сайте www.planet-it.ru были предложены тестовые задания. В итоге – более 19000 решенных в ходе заочного онлайн-тура заданий и демо-тестов и 1000 финалистов, получивших право побороться за звание лучшего в очном туре.

Пожалуй, самым дорогим подарком для студентов-победителей Олимпиады стало признание их навыков и умений высокими судьями, представлявшими потенциальных работодателей, а также возможность пройти производственную практику в региональных ИТ-компаниях. Именно они, при поддержке Intel, Microsoft, D-link, Oracle, 1С, LinuxCenter.Ru, Adobe Systems, разработали конкурсные задания региональных очных туров. В их основу легли проблемы, с которыми сталкиваются профессионалы в сфере ИТ в своей каждодневной работе.

Столь масштабное проведение IT-Олимпиады по всей России в 2009 году стало возможным благодаря беспрецедентной поддержке региональных Ассоциаций информационных технологий: АПКИТ, АКЦИТ, РАКС, АКСИТ, ТОМКИТ и ДВ АПИТ, а также органов исполнительной власти субъектов РФ.

«В январе 2010 года будет сформирована база данных перспективных студентов, которую мы предоставим региональным работодателям, кадровым агентствам, органам государственной власти, – комментирует Сергей Шалашный, председатель центрального Оргкомитета «IT-Планеты 2009". – Также в самом начале будущего года победителям Олимпиады предстоит пройти собеседования в ведущих ИТ-компаниях, после которых им будут предоставлены места для практики с возможным последующим трудоустройством. Очень надеюсь, что именно эта практика станет началом их блестящей карьеры».

› В «IT-Планете 2009» приняли участие более 6300 человек, 1000 из которых вышли в финал.



Пока гром не грянет...

Нужны ли в Linux настольные антивирусы – хоть коммерческие, хоть свободные? Последние события заставляют взглянуть на этот вопрос по-новому.

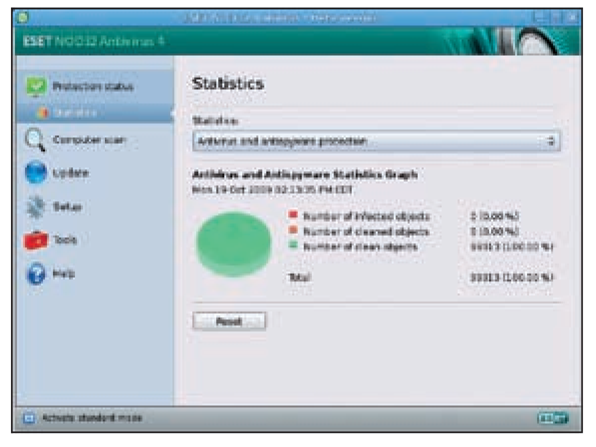
Растущая популярность Linux у пользователей заставляет активизироваться разработчиков коммерческих антивирусных пакетов. Компания ESET выпустила бета-версию *NOD32* для дистрибутивов GNU/Linux, совместимых со спецификацией LSB 3.1. Программа доступна в виде RPM- или DEB-пакетов, требует ядра версии 2.6.и установленного в системе *GTK+ 2.6*. Интересно, что *ESET NOD32 Antivirus 4 for Linux Desktop* предназначен для защиты рабочих станций. По заявлению создателей, пакет позволит блокировать специфичное для Linux вредоносное ПО, а также программы, предназначенные для Windows и Mac OS X, полученные по электронной почте или скачанные с файлообменных сетей.

Честно говоря, эффективность подобных программных продуктов у меня вызывает большие сомнения (практика показывает, что они и в Windows помогают слабо). *NOD32* изначально разработан для Windows и предназначен для борьбы с «заразой», написанной под эту систему. О каком Linux-специфичном вредоносном ПО говорят представители ESET, вообще непонятно – на сегодняшний день ни одного вируса для Linux в «диком» виде не отловлено. В свое время мне попадались исходные тексты пары таких «вредителей», но собрать их с ходу не получилось из-за проблем совместимости версий библиотек и компилятора *gcc*.

Принято считать, что от вирусов нас спасает более строгое разграничение прав пользователей, наличие проблем совместимости двоичного кода с установленной версией ядра и библиотек, а также мандатный контроль доступа. Позволю себе с этим не согласиться. Чисто технически, реализовать вирус для Linux намного сложнее, чем для Windows, где пользователь обычно работает с правами администратора системы. Тем не менее, Unix-подобные ОС тоже уязвимы для вредоносного ПО (для примера, возьмите Mac OS X). Проблемы двоичной совместимости вирусописателям помогут разрешить языки высокого уровня, позволяющие создавать переносимый код. Канал распространения? Первое, что приходит в голову – уязвимости в web-браузерах, дающие возможность удаленно выполнить на целевой системе любую вредоносную программу. Отсутствие прав администратора также не помеха: и в контексте непривилегированного пользователя можно сделать немало. Еще есть различные сторонние каталоги приложений и сервисы сборки (Launchpad, openSUSE Build Service и так далее), которые могут быть использованы злоумышленниками. В декабре на портале www.gnome-look.org было зарегистрировано

> «Пользователям стоит избегать ПО из недоверенных источников.»

появление первого «трояна» в Deb-пакете с хранителем экрана *Waterfall*. Вирусом это назвать еще нельзя – пользователь должен установить программу самостоятельно. Однако особой практической разницы я не вижу – сторонние каталоги весьма популярны и легко могут стать «переносчиками инфекции». Согласитесь, у многих линуксоидов на машинах установлен картинки или свежие версии приложений, взятые с подобных ресурсов.



› **Вирусы, ау! Даже на компьютерах разработчиков среди более 99 тысяч объектов не удалось найти ни одного зараженного.**

Так, значит, нам остается только покупать антивирусные пакеты за свои собственные деньги? Надеюсь, Linux не свернет на эту дорожку. Есть другой путь – совершенствование безопасности операционной системы. Потенциально уязвимые приложения должны запускаться в надежно изолированных «песочницах», а строгий мандатный контроль доступа – применяться повсеместно. Создателям каталогов приложений стоит ужесточить контроль за их содержимым, а пользователям, по возможности, избегать установки ПО из недоверенных источников. На первый взгляд – вещи очевидные. Но, боюсь, таковыми они станут только после построения первого ботнета из Linux-компьютеров. LXF

Новости короткой строкой

» Свободный видеодрайвер nouveau будет интегрирован в Ubuntu 10.04 – ранее о его поставке заявил проект Fedora. Этому будет способствовать включение необходимой инфраструкутуры в ядро 2.6.32. Позиция компании NVIDIA в данном вопросе остается неизменной: она будет развивать собственный закрытый драйвер и участвовать в разработке nv.

» Марк Шаттлворт [Mark Shuttleworth] оставит пост генерального директора Canonical и сосредоточится на улучшении Ubuntu. Кресло директора займет один из заместителей Шаттлворта – Джейн Зильбер [Jane Silber].

» Компания Google выпустила по открытой лицензии исходные коды *EtherPad* – текстового редактора для совместной работы через Web, который будет использоваться в Google Wave.

» Gnome может выйти из состава GNU Project – руководители проекта конфликтуют с Ричардом Столлменом.

» Microsoft разъяснила юридический статус Moonlight – любой разработчик может включить продукт в свой дистрибутив GNU/Linux. Одновременно с этим компания установила для разработчиков лицензионные отчисления за использование ExFAT в размере $300 000.

LINUX FORMAT

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов



АЛЕКСЕЙ ФЕДОРЧУК
Его слабости — mass storage, разметка диска и файловые системы.

Дистрибутив Linux: кто он сегодня?

Ответить на этот вопрос не так-то просто: большинство критериев, некогда определявших самостоятельность дистрибутива, ныне утратили свое значение. Собственная программа установки – когда-то непременный атрибут любой уважающей себя системы – уже таковым не является: удачные инсталляторы либо напрямую наследуются клонами и дериватами, либо оказывают определяющее влияние на системы даже не родственные.

Аналогично и со средствами управления пакетами: лучшие их представители, зародившись в недрах одного дистрибутива, начинают свое победное шествие не только по системам смежным, но и по «более иным» операционкам.

Комплектация программами? Скажем, когда-то Slackware содержал большое количество средств разработки, Red Hat – системного администрирования, а Mandrake – пользовательских приложений. Ныне же каждый «большой» дистрибутив включает практически весь набор свободного ПО, созданного человечеством. А удобные средства управления пакетами позволяют легко превратить самую рассерверную инсталляцию в настольную. Да не с дистрибутивного носителя, а... откуда? Правильно, из репозитория.

Вот мы и подошли к тому, что определяет своеобразие дистрибутива: наличие репозитория пакетов как системной целостности, включая средства его поддержки, обновления и наращивания. Если у тебя это есть, ты – дистрибутив. Нет – ремикс или респин...

alv@posix.ru

Сегодня мы рассматриваем:

OpenSUSE 11.2 с. 12

› Удастся ли Novell вернуть KDE славу самого уютного и удобного рабочего стола для Linux?

SheevaPlug с. 13



› Мал, да удал: полноценный Linux-сервер теперь можно уместить если не в спичечном коробке, то уж точно в корпусе от блока питания.

Наш вердикт: Пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибалльной шкале (0 – низшая оценка, 10 – высшая). Как правило, учитываются функциональность, производительность, простота использования и цена, а для бесплатных программ – еще и документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.



Выдающиеся решения могут получить престижную награду «Top Stuff». Номинантами становятся лучшие из лучших – просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но если разработчики рекомендуют *Autopackage*, мы следуем этому совету.

LINUX FORMAT Вердикт

Google Earth

Разработчики: Google
Сайт: http://earth.google.com
Цена: Бесплатно по закрытой лицензии

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Оправданность цены	9/10

» Если весь мир — сцена, то Google Earth — театр. Простая в использовании, захватывающая и ободряюще практичная программа.

Рейтинг 9/10

Ubuntu 9.10



Прихватив охапку эвкалиптовых листьев для приманки, **Грэм Моррисон** разыскивает в кустарнике нового экзота – кармического коалу.

Вкратце

» **Легендарный дистрибутив, ставший синонимом Linux на BBC. См. также: Fedora, Linux Mint, OpenSUSE, PCLinuxOS, Mandriva, Debian.**

Наши постоянные читатели знают, что мы не в восторге от коричневой гаммы Ubuntu. Но люди ее любят, да и деревья ее придерживаются, и с этим нельзя не считаться.

Вот так изящно мы подошли к медовым переливам Ubuntu 9.10 (он же Karmic Koala), визуально – самой приятной версии из виденных нами. От опрятного инсталлятора, потчующего вас информацией во время установки, до стандартного рабочего стола Gnome с обновленным комплектом тем, Karmic Koala выглядит впечатляюще.

А вот найти новые функции будет посложнее. Gnome 2.28 представлен в финальной версии; отметим замену клиента мгновенных сообщений *Pidgin* на *Empathy*. Экран входа в систему, цвета шоколада, полностью обновлен: теперь имя и пароль можно вводить на экранной клавиатуре, не отпуская привычной мыши.

> «Сравнив Koala с Warty Warthog, мы бы заплясали.»

Самое заметное дополнение к этой версии (заслуга Canonical) – Центр приложений Ubuntu [Software Centre], новый значок в самом низу меню приложений. Кажется, пример Apple с его iPhone и AppStore оказался заразительным.

Пока это только замена функции добавления и удаления программ (присутствовавшей в предыдущих версиях), но есть планы расширить концепцию за счет подключения коммерческих приложений. Доступен поиск пакетов по категориям или в реальном времени, через поисковое поле. Но здесь нет ни рейтингов, ни оценки дискового пространства для пакета, ни средств просмотра пользовательских отзывов – словом, проку от такого дополнения пока немного.



› **Новый сезон принес свежие оттенки коричневого в палитру Ubuntu. Если закрыть глаза на цвета, мы считаем Karmic Koala приятнейшим из всех Ubuntu.**

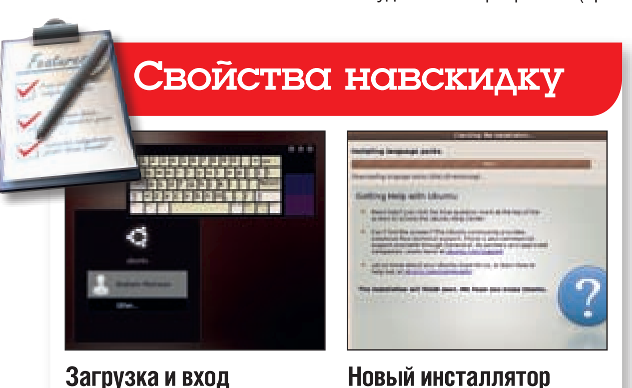
Облака

Ubuntu One, «персональное интернет-облако» Canonical, теперь тесно увязано с рабочим столом. Щелкните по его значку – вам будет предложено ввести имя пользователя и пароль Launchpad, и Ubuntu One скроется в фоне. Оттуда приложение синхронизирует все файлы, находящиеся в каталоге **Ubuntu One** вашей домашней директории. Синхронизируются также контакты и заметки *Tomboy*, и на любой машине, где вы станете работать с той же учетной записью, синхронизация выполняется автоматически. Несмотря на бета-состояние, функция весьма перспективна для тех, у кого есть несколько компьютеров.

Но самое значительное новшество в версии 9.10 – это изменения в процессе загрузки. Обязанности загрузчика переданы *Grub 2.0*, что обеспечивает родную поддержку идентификации устройств по UUID и скриптов, а также более функциональное графическое меню. Файловой системой по умолчанию для Ubuntu стала ext4, улучшающая производительность. Процесс загрузки контролирует *Upstart*, благодаря чему время на нее уменьшилось со 100 секунд (Jaunty) до 41 секунды. Уже одно это – достаточно веская причина для обновления.

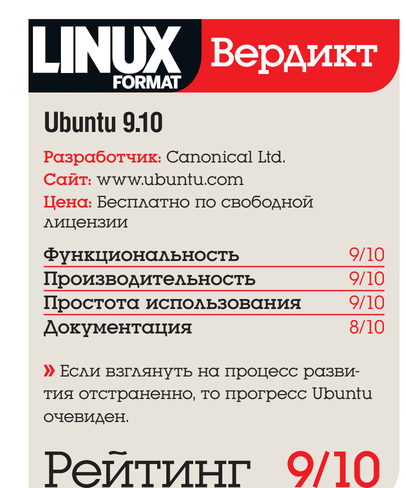
Похоже, Canonical перераспределяет в Ubuntu акценты, ради новых источников доходов, но по-прежнему продолжает вести мяч прогресса. Сравнив Koala с Warty Warthog, мы бы заплясали; увидев небрежение Canonical, заворчали бы. А так, Karmic Koala – трезвый, надежный и безопасный дистрибутив, и мы ему тихо радуемся. LXF

Свойства навскидку



Загрузка и вход
Время загрузки до приглашения значительно уменьшилось, а сам экран полностью обновлен.

Новый инсталлятор
Пока Ubuntu устанавливается, вы можете изучить коллекцию полезных «шпаргалок».



LINUX FORMAT Вердикт

Ubuntu 9.10

Разработчик: Canonical Ltd.
Сайт: www.ubuntu.com
Цена: Бесплатно по свободной лицензии

Функциональность	9/10
Производительность	9/10
Простота использования	9/10
Документация	8/10

» Если взглянуть на процесс развития отстраненно, то прогресс Ubuntu очевиден.

Рейтинг 9/10



Mandriva 2010.0

Вроде и нелепо выпускать дистрибутив вослед новому Ubuntu, но **Маянк Шарма** нашел француза серьезным соперником африканцу.

Вкратце

» Один из основных дистрибутивов Linux, выстроенный вокруг KDE. См. также: Ubuntu или Fedora.

Ключевое ПО

- » Ядро 2.6.31.5
- » Gnome 2.28.1
- » KDE 4.3.2
- » OpenOffice.org 3.1.1
- » Сервер X.org 1.6.5

Не прошло и недели после того, как наш рабочий стол украсил Karmic Koala, а нам пришлось заново разбивать диск – под Mandriva 2010.0. Честно говоря, выпуск новой версии так скоро после популярного дистрибутива выглядел не самым разумным ходом Mandriva. Но последняя версия Ubuntu больше разочаровала пользователей, чем порадовала, и Mandriva оказался весьма кстати.

Для новичков, в Mandriva реализация KDE не только свежее, но и функциональнее, чем в Kubuntu. Вложен немалый труд, чтобы вернуть KDE 4 вид и стиль поведения KDE 3. Разворачиваемые меню отключены, значки на рабочем столе присутствуют «из коробки». Kubuntu, наоборот, завел пару новых виджетов, лишний раз загромождающих ценную площадь экрана.

Хотя Gnome не является основным рабочим столом Mandriva, дистрибутив извлек из версии 2.28 те же преимущества, что и Ubuntu, включая переход на клиент мгновенных сообщений *Empathy*.

> «Из Grub в KDE мы попали менее чем за 15 секунд.»

Как обычно, Mandriva выходит в нескольких ипостасях: бесплатный One на Live CD с возможностью установки, с Gnome и KDE; Free DVD без «несвободных» драйверов и приложений; и проприетарный Powerpack, распространяемый за деньги и снабженный кое-каким коммерческим ПО. Мы будем ориентироваться на One, как основной способ знакомства с Mandriva для новых пользователей.



› Хотя в комплект входят и Gnome, и KDE, последний остается фаворитом Mandriva: здесь лучше и поддержка мультимедиа, и интеграция.

Как и Ubuntu, Mandriva работает над сокращением времени загрузки: на нашей машине с процессором Intel Core 2 Duo E6300 и 2 ГБ ОЗУ мы попали из *Grub* в KDE менее чем за 15 секунд.

Всегда готов!

Сразу после инсталляции, рабочий стол KDE из редакции One гораздо более работоспособен, чем в большинстве других крупных дистрибутивов. Нашлись драйверы и для адаптера Linksys USB, и для видеокарты Nvidia, которые Mandriva немедленно берет в оборот. Медиа-система *Dragon Player* позволяет воспроизводить файлы AVI, MPEG, DivX и MKV; кроме того, есть медиа-центр *Moovida*. В качестве офисного пакета в Mandriva применяется *Go-OO* (ответвление *OpenOffice.org*).

Как и Ubuntu One, Mandriva предлагает возможность хранения данных в «облаке», но берет за это плату. Деньги делаются и на распространении USB-брелков с дистрибутивом, а в магазине Mandriva продается оборудование, совместимое с Linux.

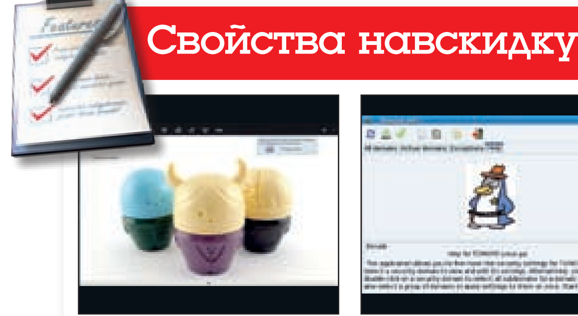
Пользователей нетбуков порадует новейшая рабочая среда Moblin v2, а вот для *K3b* на Live CD места не нашлось.

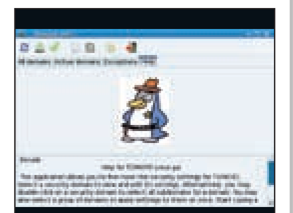
Если говорить о параметрах настройки, то новая версия включает каркас безопасности *Tomoyo Linux*, а менеджер сетевых профилей полностью переписан и оснащен новым монитором сетевой активности. Инструмент родительского контроля получил новый таймер и возможность составления «черного списка» программ.

В наши дни разбивка диска – уже не головоломка, и инсталлятор One без труда пододвинет существующие разделы Windows, выделив место для Mandriva. Перед копированием файлов удаляются лишние драйверы и языковые пакеты: дисковое пространство нужно беречь. Существует гостевая учетная запись с ограниченными правами, которая позволяет постороннему человеку исследовать дистрибутив, не покушаясь на ваши файлы или настройки.

Хотя дистрибутив ориентирован на настольное применение, программисты тоже не забыты. Mandriva 2010.0 укомплектован *Apache 2.2.14, PHP 5.3,* Python 3 и другими инструментами для разработчиков. LXF

Свойства навскидку





Поддержка нетбуков

Помимо среды Moblin v2 для нетбуков, новая версия поддерживает драйверы Poulsbo прямо «из коробки».

Новая безопасность

В новом дистрибутиве система *AppArmor* замещена на *Tomoyo Linux*, на сей раз с графическим интерфейсом.

LINUX FORMAT Вердикт

Mandriva Linux 2010.0

Разработчик: Mandriva
Сайт: www.mandriva.ru
Цена: бесплатно

Функциональность	7/10
Производительность	8/10
Простота использования	8/10
Документация	7/10

» Если вы пользователь KDE, возьмите дистрибутив в новый год. А вот Gnome-версия отстает от Ubuntu.

Рейтинг 8/10

OpenSUSE 11.2



Поджарый, зеленый, опрятный и выбравший ящерицу в качестве тотема... Нет, это не **Ник Вейч**, а очередная версия открытого дистрибутива от Novell.

Вкратце

» Один из большой тройки (четверки?) дистрибутивов Linux. Ближе к Mandriva, чем к Ubuntu.

Ключевое ПО

- » Ядро 2.6.31.5
- » KDE 4.3.3
- » Gnome 2.28
- » OpenOffice 3.1.1
- » Firefox 3.5.4
- » GCC 4.4

В последней версии OpenSUSE заметен прогресс инсталлятора – он стал удобнее, привлекательнее и функциональнее. Вообще-то он и был неплох, но улучшения заметны: например, к началу стандартной установки вы теперь перейдете за какие-то минуты. Инсталлятор поддерживает различные устройства ввода, легко управляется с разбивкой диска и, что особенно важно, обеспечивает такой уровень детализации, при котором отформатировать «не то» устройство практически невозможно.

Установочный DVD можно применять для тестирования драйверов и восстановления системы, но чтобы воспользоваться этими функциями, нужно быть на короткой ноге с *Bash*.

Обычная проблема при установке – в комплекте поставки отсутствуют пакеты, изрядно облегчающие жизнь пользователю. Но это не вина SUSE: такие вещи, как микропрограммы и проприетарные модули (Flash, это про вас!), нельзя включать из-за лицензионных ограничений. Задача

> «К началу стандартной установки вы перейдете за какие-то минуты.»

дистрибутива – по возможности восполнить нехватку после инсталляции. Команда SUSE постаралась убрать все препоны на этом пути. Например, драйверы Nvidia добавить реально легко. Увы, легкость начнется только после того, как вы разыщете инструкции по установке на странице поддержки web-сайта, да и сам сайт. Что, конечно, плоховато: многие могут попросту не знать о том, что доступен более подходящий драйвер, и не станут его и искать.

› KDE 4.3.3 во всей красе: тема «зеленое поле» с примесью, видимо, одуванчиков. А не нравится KDE 4 — у OpenSUSE наготове Gnome.

По тем же причинам, сразу после установки полезно провести обновление. Скажем, насущно необходимые шрифты MS Core Fonts на DVD отсутствуют, а обновление это исправит.

На рабочем столе

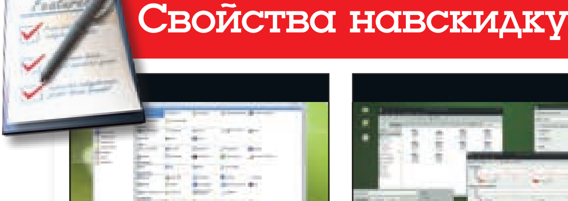
Последние версии KDE получили неоднозначную оценку. Вероятно, поэтому OpenSUSE особо подчеркивает возможность работы с новейшей версией Gnome (2.28). OpenSUSE «дружит» с обоими рабочими столами, и все найдется там, где ему положено быть, без особых хлопот. Хотя OpenSUSE, пожалуй, всех превзошел в реализации KDE, ощущение незавершенности все же остается. Впрочем, если вы уже привыкли к KDE 4.x, вам и карты в руки.

Gnome 2.28 попал в руки к SUSE едва ли не перед самым выпуском, но за оставшееся время его успели достойно вписать в дистрибутив. В обеих рабочих средах стандартный браузер по-прежнему *Firefox 3.5.4* (любимый многими), но, за исключением этой и нескольких других областей, «умолчания» в OpenSUSE для разных рабочих столов разные. Кроме, разумеется, *Slab*, «запускалки» программ OpenSUSE, набирающей популярность. *Yast*, пожалуй, интегрирован слишком хорошо – так, что почти неотличим от Центра управления Gnome, хотя оба выполняют одни и те же функции с помощью разных приложений.

Легкий реверанс в сторону микроблогов – по умолчанию устанавливается *Choqok*, комбинированный настольный клиент для Identica и Twitter с рядом полезных функций. Однако для других протоколов по-прежнему используется *Pidgin*.

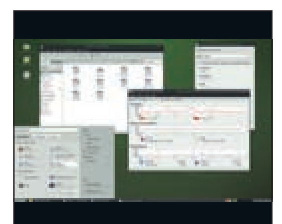
Хотя львиная доля работы над данной версией осталась скрытой «за кулисами», очевидно, что OpenSUSE держит курс на выпуск качественных, добротных дистрибутивов. LXF

Свойства навскидку



Большой Yast

Взгляд на просторы *Yast* (не путать с Центром управления Gnome). Еще чуток параметров – и придется докупать новый монитор.



Поддержка Gnome

Gnome у OpenSUSE хоть и не фаворит, но поддержкой не обижен (с некоторыми примочками типа *Slab*).



Вердикт

OpenSUSE 11.2

Разработчик: Novell и сообщество OpenSUSE
Сайт: www.opensuse.org
Цена: Бесплатно по свободной лицензии

Функциональность	8/10
Производительность	9/10
Простота использования	8/10
Документация	9/10

» Мировой революцией не пахнет, но дистрибутив стабильный, современный и надежный.



Рейтинг 8/10

SheevaPlug Dev Kit

Грэм Моррисон принял за блок питания... полноценную Linux-машину!

Вкратце

» Полноценный Linux-компьютер в коробочке, на которой поместится разве что чайная чашка. Аналоги: Lintop или Bubba Home Server.

Название «SheevaPlug» обусловлено, вероятно, тем, что эта белая коробочка по размеру лишь чуть больше среднего блока питания, и, снабдив соответствующим переходником, ее можно подключить прямо к домашней розетке. Но вместо преобразователя напряжения внутри расположен настоящий Linux-компьютер с флэш-памятью 512 МБ, 512 МБ ОЗУ и процессором ARM тактовой частотой 1,2 ГГц. Есть стандартный USB-порт, гнездо для карты памяти SD и сетевой Ethernet-разъем RJ45. Предустановленная ОС – новейшая сборка Ubuntu для ARM.

К названию приписано Development Kit («Комплект разработчика»), так как предполагалось, что программисты встраиваемых решений будут пользоваться подобными машинами при создании своих продуктов, распространяя результат в виде готовых приставок на базе того же самого оборудования. Это правда, но среднего пользователя может отпугнуть. По сравнению со стандартным NAS, SheevaPlug мощнее и может служить ему превосходной заменой (при наличии внешнего накопителя). Причем комплект обойдется дешевле многих специализированных NAS. Но SheevaPlug с равным успехом можно использовать и как медиа-плейер (с выводом звука через USB-совместимые колонки), и как хаб для домашней системы автоматизации, и даже как сетевой сервер времени (понадобится только GPS-модуль с USB-портом).

> «Лучшее в этой машине – ARM-порт Ubuntu.»



› **Подключаете SheevaPlug к Ethernet-кабелю — и вперед!**

Мал, да удал

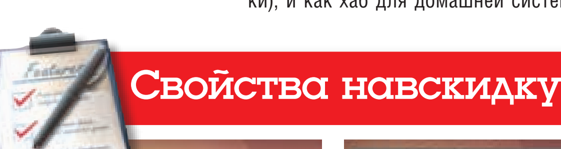
Чтобы включить SheevaPlug в работу, необходимо подсоединить другой Linux-компьютер к USB-порту, загрузить модуль FTDI, затем через эмулятор терминала подключиться к последовательному порту USB. После этого вы сможете войти в SheevaPlug и управлять им через командную строку. Это JTAG- и консольный интерфейс устройства. На стандартной модели, прежде чем начинать работу, необходимо ввести несколько команд для изменения прав доступа, отредактировать параметры DNS и обновить пакеты дистрибутива, но более 10 минут это не займет. В итоге вы получите чистую систему и доступ по SSH.

Лучшее в этой машине – то, что в ней используется новый ARM-порт Ubuntu. Благодаря ему можно установить едва ли не любое приложение, через **apt-get install**. Быстрая прикидка в командной строке показала 25 292 доступных пакета, что сравнимо с возможностями типичного NAS.

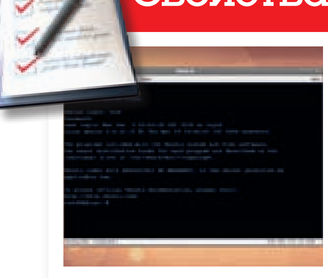
Тестируя устройство, мы загрузили несколько увесистых приложений, включая *Apache*, *MySQL*, PHP 5 и, наконец, *WordPress* – инсталляция прошла безупречно. Мы следовали стандартным рекомендациям для Ubuntu и подивились эффективности крохотной коробочки (хотя Slashdot-эффект она бы не выдержала).

Кроме того, мы установили *Squeezebox*, стандартный сервер для вещания музыки по сети. Он показался нам необычайно быстрым, гораздо быстрее устройств от Synology и QNAP. Похоже, SheevaPlug может служить идеальным музыкальным центром, да и *MediaTomb* здесь действует не хуже.

Малые размеры, низкое энергопотребление и дешевизна делают SheevaPlug идеальной платформой для экспериментов. Он, например, является прекрасной заменой для снятого с производства Linksys NSLU2. Существует солидная онлайн-поддержка для расширения объема памяти, загрузки с USB- и флэш-устройств, и много других интересных проектов. Об удобстве использования, конечно, говорить не приходится, но тот, кто знает, куда пристроить SheevaPlug, в убеждении не нуждается. LXF

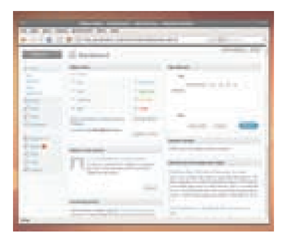


Свойства навскидку



Отладочная консоль

Работает машина или нет – неважно: эмулятор терминала для подключения к консоли доступен всегда.



Ubuntu

Можно установить тысячи пакетов, не беспокоясь о возможностях процессора или зависимостях.



LINUX FORMAT Вердикт

SheevaPlug

Разработчик: NewIT Ltd
Сайт: http://newit.co.uk
Цена: £89 (около 4500 рублей)

Функциональность	8/10
Производительность	8/10
Простота использования	6/10
Оправданность цены	9/10

» Недорогой, стабильный, гибкий в настройке и мощный. Как Linux.

Рейтинг 9/10

Сравнение

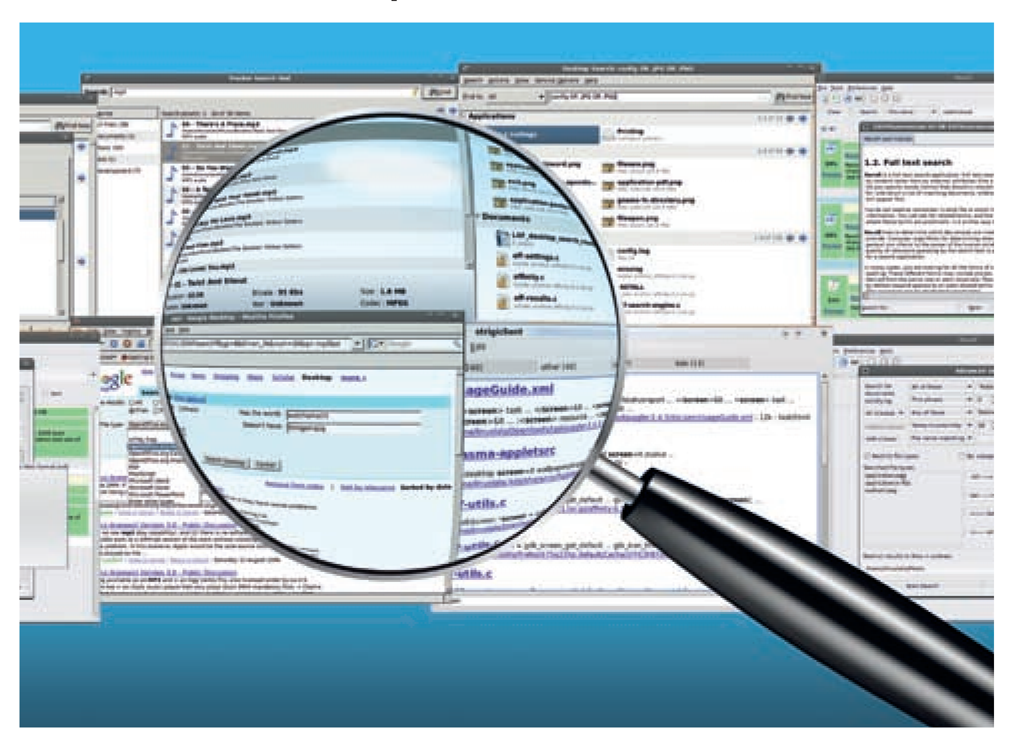


Каждый месяц мы сравниваем тысячи программ – а вы можете отдыхать!

Настольные поисковики

Ознаменуют ли эти продвинутые инструменты кончину *grep*? Выясняет **Шашанк Шарма**.





Про наш тест...

Трудности установки и завышенные требования к памяти – не то, чем характерны поисковики нашего теста: 512 МБ ОЗУ хватило большинству из них. Поскольку размер индекса растет с числом файлов, нам нужна утилита, умеющая искать все необходимое быстро и точно.

Насколько допустима расплывчатость критерия поиска, чтобы нужный файл все же нашелся? С другой стороны, сколько информации позволяет ввести программа, если вы хотите быть максимально конкретными? Бонусные очки достаются приложениям, сохраняющим размер индекса небольшим и предлагающим вариации поиска, например, ограничение результатов определенными типами MIME. Классная утилита немыслима без морфологического поиска, чтения метаданных, доклада о необычных форматах и поиска текста внутри полей.

Наш выбор

Инструменты типа *grep*, *find* и *awk* часто выручают бесшабашных любителей *Bash* при поиске файлов, зарытых под посторонними гигабайтами. Но поскольку типичный дистрибутив Linux и сам замахивается на гигабайты дискового пространства, нетрудно представить, что поиск файлов со временем только усложнится.

Как и их популярные интернет-собратья, современные инструменты поиска на рабочем столе могут не только находить файлы по именам, но и выполнять контекстно-зависимый поиск внутри архивов электронной почты, изображений, видео и музыки. Некоторые поисковики заходят дальше, и даже индексируют историю вашего браузера и закладки.

> «Поисковики заходят дальше, и индексируют историю браузера.»

Настольные поисковики работают через созданный индекс всех файлов в вашей системе. Когда пользователь ищет файл, такой инструмент, вместо просмотра всего диска, лишь пробегает по индексу. Поскольку существование настольных поисковиков оправдывается необходимостью искать файлы быстрее и удобнее, чем с помощью обычных инструментов вроде *find* и *grep*, они должны быть быстрыми и надежными и предлагать максимум информации, чтобы помочь вам быстро находить требуемые объекты.

Поэтому большинство поисковиков умеют читать метаданные файлов, предоставлять выдержки из текста документа, показывать разрешение изображений (а также миниатюры) и выводить другие детали. Стемминг – то есть если вы ищете «бит», инструмент находит соответствие с «Битлз», «битый» и т.д. – это стандарт.

Для большинства настольных поисковиков характерен удобный графический интерфейс, но многие поставляются также и с набором приложений командной строки для индексирования и поиска по файловой системе.

Beagle

Все еще носит клеймо пожирателя памяти, но становится классикой.

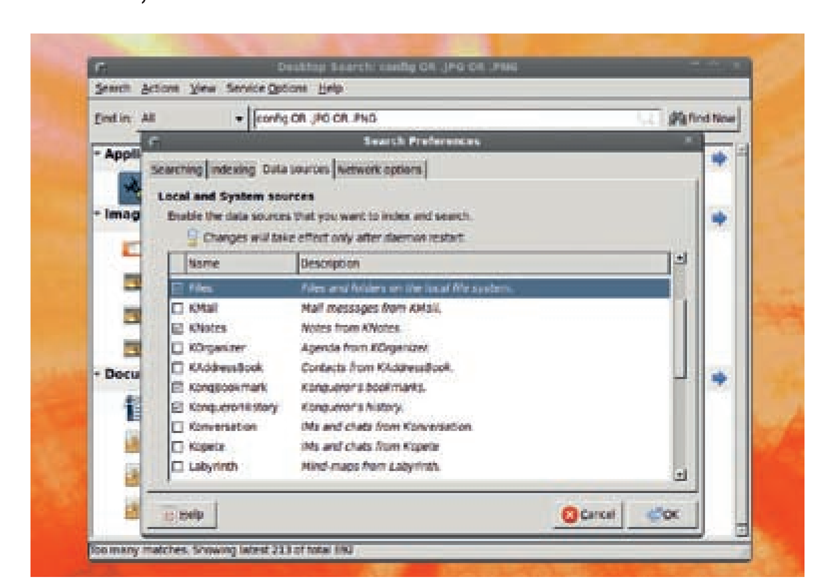
Дискуссиям о привычке *Beagle* к пожиранию памяти, длящимся с давних славных дней Ричарда Бертона [Richard Burton], пора бы прекратиться. Так или иначе, это яркий пример приложения, которому мешают ответшалые ярлыки, хотя бы и неточные. Стоит ли удивляться, что обильные сообщения на форумах и блогах о его ненасытном аппетите по части памяти распугали немало потенциальных пользователей? Последняя версия, 0.3.9, доступна в репозиториях почти всех дистрибутивов.

Контроль над тем, какие папки индексировать и какие пути игнорировать, стал стандартом в большинстве настольных поисковиков, и здесь *Beagle* не разочаровывает. Зато, в отличие от большинства других инструментов, он также позволяет индексировать электронную почту, мгновенные сообщения, RSS-ленты, адресную книгу и многое другое, да еще и историю серфинга и закладки вашего браузера. А со своей встроенной поддержкой *Inotify*, *Beagle* обновляет индекс при обнаружении любых изменений в файлах или каталогах.

По умолчанию, *Beagle* индексирует все содержимое вашей домашней директории, кроме ***~**, **~.tmp** и тому подобных путей. Чтобы изменить это поведение, запустите *Beagle*, обычно значащийся как Поиск [Search] в меню Приложения > Дополнительно [Application > Accessoires], и перейдите в Поиск > Настройки [Search > Preferences]. Из Индексирования [Indexing] окна Настройки поиска окна можно указать охватываемые утилитой каталоги, а также исключаемые пути.

Наряду с графическим интерфейсом, *Beagle* имеет обширный набор инструментов командной строки, пригодных для создания индекса и поиска файлов. Команда **beagle-search .txt** запускает графический интерфейс и показывает результаты поиска для **.txt**. Альтернатива – запуск **beagle-query**, для вывода результатов прямо в терминал.

› Использование шаблонов и операторов OR и AND помогут быстрому поиску файлов.



Интерфейс браузера

Beagle чувствует себя одинаково хорошо и под Gnome, и под KDE, но если вы предпочитаете нейтральное окружение, это можно легко организовать. Для запуска сетевого интерфейса, откройте окно терминала и введите следующую команду:

```
beagle-config Networking
WebInterface true
```

Доступ к экспериментальному web-интерфейсу предоставляется на http://localhost:4000. Предполагалась доступность и с других компьютеров в сети, но... это пока эксперимент.

При использовании настольных поисковиков помните, что лишь немногие из них отличают имена файлов от типов, так что запросы "mp3" и ".mp3" дадут весьма непохожие результаты.

Beagle может извлекать текст и метаданные из множества типов файлов, включая документы *Office*, простой текст, HTML, DocBook, различные изображения и аудиоформаты, и многое другое. При поиске файлов можно уточнить параметры запроса в одной из 14 доступных категорий: фотографии, медиа, файлы, архивы, письма и т. д. Выберите тип поиска в выпадающем списке, чтобы сузить критерии. *Beagle* отображает всего восемь элементов на странице, и лучше максимально уточнить критерии поиска, чем щелкать по крошечной синей стрелке навигации, надсажая свою мышь.

Если при поиске категория не указана, *Beagle* все равно рассортирует результаты, сгруппировав вместе изображения, документы, папки и т. д. При избытке результатов вы затем сможете перемещаться по страницам в каждой из категорий.

Beagle также умеет выполнять поиск внутри файлов. Щелчок по найденному файлу покажет вам совпадение или кусок текста, имеющий отношение к делу. Это не относится к PDF-документам, для которых *Beagle* даст вам только миниатюру.

Интерфейс *Beagle* малоэстетичен, но настоящая красота кроется в его комплексных опциях поиска. Можно, например, предварять термины знаком «минус», чтобы исключить их из своего запроса, или использовать в запросе оператор OR (ИЛИ), или оператор date для проведения поиска некотором в диапазоне дат.



› *Beagle* умеет индексировать данные разнообразных приложений, включая клиенты обмена мгновенными сообщениями и ленты новостей.



Beagle

Версия: 0.3.9
Сайт: http://beagle-project.org
Цена: Бесплатно под GPL

» Выживает, несмотря на плохой пиар; небольшой редизайн интерфейса привел бы *Beagle* к победе.

Рейтинг 9/10

Google Desktop

Фамильярность порождает уверенность. Но работает ли это?

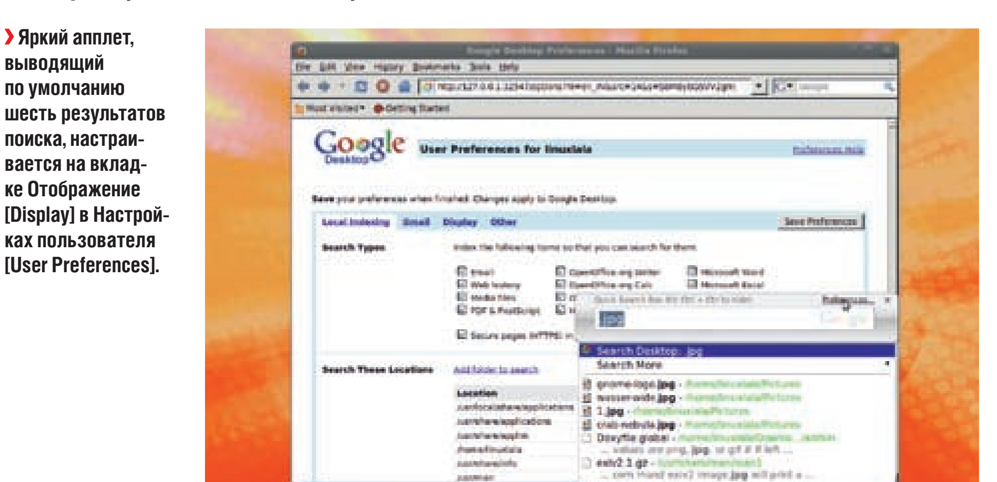
Для большинства дистрибутивов релиз-циклы – это все; но *Google Desktop*, кажется, был доступен через интернет-репозитории ПО вечность назад. Сейчас вам нужно всего лишь направиться на домашнюю страницу проекта и загрузить последнюю версию. Там даже найдется 64-разрядная редакция. Мы добавим *Google Desktop* очков и за предоставление двоичных пакетов в форматах RPM и Deb.

После установки *Google Desktop* садится в системный лоток и немедля берется за индексацию. Хотя индексирование диска не особо отражается на системных ресурсах, оно все же требует времени. Настройте программу по своему вкусу с самого начала, определив папки для индексирования и какие типы файлов следует игнорировать.

Google Desktop также может следить за вашей историей серфинга и всеми вашими учетными записями электронной почты, благодаря очень хорошей *Thunderbird*. Вы можете настроить его и на индексирование учетной записи Gmail, даже если вы заходите туда через браузер.

Удобный апплет поиска вызывается двойным нажатием клавиши Ctrl – и вы можете поместить его где-то на вашем рабочем столе для быстрого доступа. При наборе слова в поисковом апплете результаты запроса будут отображаться там же. Клавиши стрелок переместят вниз к искомому файлу; чтобы открыть его, нажмите Enter. Затем вы можете нажать на Просмотреть все результаты в браузере [Show All Results In A Browser], чтобы открыть новую вкладку *Firefox*, если он запущен. Если нет, *Google Desktop* откроет *Firefox* для вас.

Google предлагает массу различных сервисов, таких как Группы, Карты и Новости, и можно настроить апплет для поиска указанного термина в любом из них. Нажмите правой кнопкой мыши на иконке в системном лотке и выберите Тип поиска по умолчанию [Default Search Type] в Интернете, на Компьютере, в Новости, Группах, Изображениях и даже Мне повезет.



› Яркий апплет, выводящий по умолчанию шесть результатов поиска, настраивается на вкладке Отображение [Display] в Настройках пользователя [User Preferences].

При поиске на рабочем столе результаты отображаются в браузере, как при обычном поиске Google. Небольшая иконка слева от каждого результата отражает тип файла, чтобы вы не путали свои MP3 с электронными письмами. Для текста и PDF-файлов под результатом приводится небольшой отрывок. Нажатие на файлы приведет к их открытию в подходящих приложениях. Письма открываются в самом браузере, но вы также можете использовать Ответить через Gmail [Reply With Gmail] и Читать в Gmail [Read In Gmail].

Мне повезет

Расширенный поиск выполняется щелчком правой кнопки мыши на иконке приложения и выбором Показать домашнюю страницу [Show Home Page]. Затем щелкните по ссылке Расширенный поиск [Advanced Search]. Поиск можно ограничить заданным типом файлов. Например, для поиска ODT-файлов активируйте радиокнопку Файлы [Files] и выберите *OpenOffice.org Writer* из выпадающего списка.

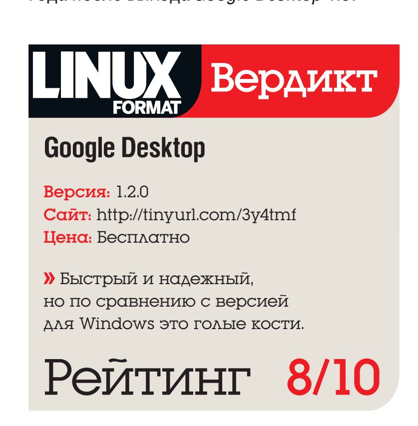
› Сузьте ваш поиск, выбрав Расширенные опции в *Google Desktop*.

Несмотря на умный дизайн, Расширенный поиск имеет одну проблему: его нельзя использовать, чтобы увидеть список всех файлов определенного типа. Невзирая на категории и типы, всегда нужно вводить поисковый запрос. И если вы хотите получить список всех PNG- или MP3-файлов на вашем компьютере, просто задайте в поиске ".mp3", не заходя в раздел Расширенный поиск.

В *Google Desktop* есть функция, которой лишены его конкуренты: в Google она называется «версией», и мы считаем, что это блестящая идея. При каждом редактировании файла, *Google Desktop* создает и кэширует копию, и к ней можно получить доступ. Нажмите на ссылку Сохранено в кэше [Cached] в правом нижнем углу результата, чью кэшированную версию надо просмотреть. Отобразятся все сохраненные версии; самая новая будет наверху.

Google Desktop можно настроить так, чтобы он больше не индексировал удаленные файлы, но тогда утратится шанс спасти данные, стертые по ошибке.

Раз уж *Google Desktop* также доступен на Windows и Mac, логично сравнить характеристики на разных платформах. Хотя Linux-версия работает безупречно, имеет удобный интерфейс и умеет индексировать электронную почту и историю браузера, немного обидно, что она лишена роскоши проприетарных ОС, даже спустя два года после выхода *Google Desktop 1.0*.



LINUX FORMAT **Вердикт**

Google Desktop

Версия: 1.2.0
Сайт: http://tinyurl.com/3y4tmf
Цена: Бесплатно

» Быстрый и надежный, но по сравнению с версией для Windows это голые кости.

Рейтинг 8/10

Tracker

У этих запасливых парней из Gnome есть все.

Вот еще один инструмент настольного поиска – и с примочкой для вашего системного лотка, но только если вы работаете под Gnome. Пользователи KDE могут поохотиться за ним в любимом K-меню. *Tracker*, как и большинство участников данного Сравнения, еще не дорос до версии 1.0, но он уже сейчас надежен, элегантен и эффективен.

Доступный в репозитариях большинства дистрибутивов, *Tracker* сперва кажется немного чудаковатым. А как иначе вы бы определили индексатор файлов, в котором индексирование по умолчанию отключено?

Открытие окон настроек и включение индексирования выполняются командой **tracker-preferences**. В отличие от других инструментов, *Tracker* настраивается на индексацию определенных папок, но не на активный их просмотр. То есть при изменении файлов, находящихся в папке, которую *Tracker* не индексирует, изменения не будут отражены в индексе.

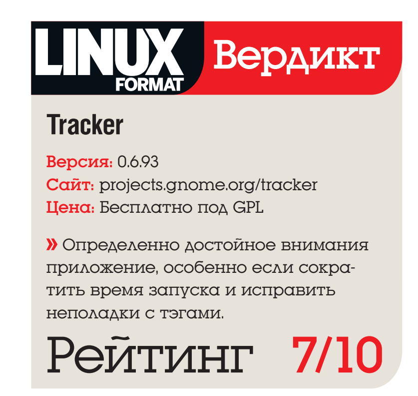
Окно настроек состоит из нескольких вкладок, каждая из которых связана с различными аспектами *Tracker*. Можно определить пути и шаблоны, которые вы хотите игнорировать в *Tracker*, во вкладке Файлы [Files], и включить индексирование электронной почты для *Evolution* на вкладке Email. Будущие версии будут поддерживать индексацию закладок браузера, истории, заметок, задач и т.д.

Отображение по умолчанию 10 результатов на странице в *Tracker* изменить нельзя, но вам помогут кнопки Следующая [Next] и Предыдущая [Previous], служащие для прокрутки страниц. При нажатии на результат *Tracker* показывает подробную информацию о файле – скажем, его размеры, если это изображение.

Tracker также позволяет добавлять к индексируемым файлам тэги. Можно использовать одинаковые тэги для разных файлов, чтобы создать коллекцию легкодоступного сгруппированного контента. Увы, это не очень надежно, поскольку иногда *Tracker* не отображает все файлы, для которых назначен один и тот же тэг.

› Результаты разделены по категориям справа. Используйте стрелки для перехода по страницам.

> «Tracker настраивается на индексацию, но не просмотр папок.»



LINUX FORMAT Вердикт

Tracker

Версия: 0.6.93
Сайт: projects.gnome.org/tracker
Цена: Бесплатно под GPL

» Определенно достойное внимания приложение, особенно если сократить время запуска и исправить неполадки с тэгами.

Рейтинг 7/10

Strigi

Тут, ребята, ловить нечего.

Самый быстрый и компактный инструмент настольного поиска (по словам разработчиков *Strigi*) нас разочаровал: мы ни разу не дождались корректного результата по своему запросу. Авось вы будете счастливее...

Устанавливается он достаточно легко, из репозиториев большинства дистрибутивов. Задуманный как графическая замена *grep* и *find*, из-за своих недостатков *Strigi* стал аутсайдером нашего списка.

Strigi вызывается командой **strigiclient** или через диалог Alt+F2. По умолчанию он не создает записи ни в каком меню, а папки начинает индексировать при нажатии кнопки Запуск индексирования [Start Indexing]. В зависимости от числа файлов, размер индекса может быть очень велик, и за ним нужно приглядывать.

Собравшись использовать *Strigi*, начните набирать слово в текстовом окне в нижней части экрана. Соответствующие результаты отображаются практически при каждом новом символе, поэтому он заслуживает звания самого быстрого инструмента поиска. Но хотя заявлялось, что *Strigi* способен искать внутри архивов, все тесты показали обратное.

Худшая проблема *Strigi* в том, что он показывает только первые 10 результатов поиска по запросу; и если найдется 55 совпадений, их список целиком вам не видать. Необходимость бесконечно уточнять параметры поиска, чтобы искомый файл попал в первые 10 результатов, уничтожает весь смысл настольного поисковика. Даже профаны в командной строке вряд ли будут менее удачливы, обратившись к *find* и *grep*.

Для инструмента, который единым духом увеличивает размер индекса до 200 МБ за полчаса, тот факт, что он неспособен найти нужный вам файл, просто ошарашивает.



› Найдет уйму файлов, но не тот, что вам нужен. Для пролистывания страниц механизма нет.

> «Strigi покажет только первые 10 результатов поиска по запросу.»

LINUX FORMAT Вердикт

Strigi

Версия: 0.6.3
Сайт: http://strigi.sourceforge.net
Цена: Бесплатно под GPL

» Не способен продемонстрировать даже заявленный на сайте функционал. Ни за что не покажет искомый файл.

Рейтинг 1/10

Recoll

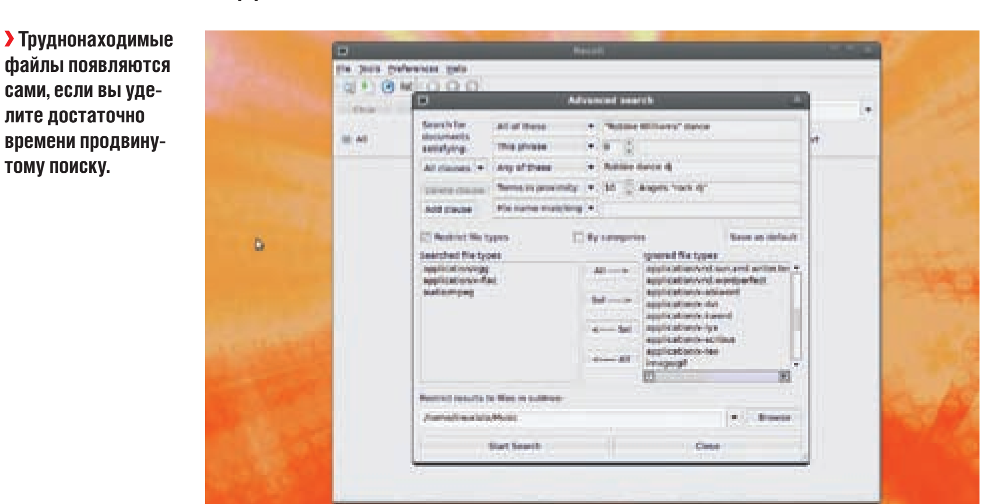
Быстрый и надежный, но не лишен недостатков.

Обладая мощью поискового движка *Xapian*, легкий *Recoll* способен стать вашим любимым инструментом, если вы сумеете установить и запустить его. В репозиториях почти всех дистрибутивов *Recoll* отсутствует, а его список зависимостей отпугнет некоторых пользователей. Для начала вам понадобится *xapian-core*, плюс *Qmake* и *Qt* – к счастью, они-то доступны легко. Быть бы у *Recoll* 10 штрафным очкам, кабы не двоичные пакеты для Ubuntu, Fedora, Mandriva и других дистрибутивов Linux.

Recoll начнет индексацию, когда вы выберете Файл > Обновить [File > Update], а индекс хранит в папке **~/.recoll/xapiandb/**. По умолчанию, индексация начинается с вашего домашнего каталога, включая примонтированные разделы или общие папки *Samba.* Дистрибутивы, использующие *Gvfs* (*Gnome Virtual File System)*, монтируют их в **~/.gvfs/**, и когда начинается индексирование домашнего каталога, разделяемые ресурсы тоже оказываются охваченными. Чтобы избежать определенных путей и каталогов, можно настроить *Recoll* через окно Prefs.

Помимо строки поиска в верхней части интерфейса, вы можете использовать ключевые слова в выпадающем списке, для выбора одного из терминов, всех терминов, имени файла и языка запроса. Поиск можно ограничить текстовыми файлами или любым другим типом MIME, щелкнув по соответствующей радиокнопке под полем поиска. Например, разыскивая сообщение электронной почты, выберите Сообщения [Messages]. При поиске файлов в индексе допускается использовать метасимволы *, ? и квадратные скобки. Это, наряду с функцией автозаполнения (доступна при нажатии Esc+пробел), дает *Recoll* небольшое преимущество по сравнению с другими инструментами. Например, выбор «pyt» и нажатие Esc+Space отобразит список возможных терминов типа Python, pytype и т. д.

› Труднонаходимые файлы появляются сами, если вы уделите достаточно времени продвинутому поиску.



Пойди найди

В зависимости от вашего поиска и количества выводимых результатов, вам, возможно, придется просмотреть несколько страниц, чтобы найти то, что вы ищете – просто нажмите на ссылку Следующая страница [Next Page] в правом верхнем углу панели результатов. При отображении результатов *Recoll* печатает рядом с каждым из них небольшой фрагмент, хотя его не всегда достаточно, чтобы решить, нашли ли вы то, что хотели. Удобно будет нажать на ссылку Предпросмотр [Preview] рядом со списком результатов и прочесть содержимое файла в программе просмотра документов *Recoll*.

Чтобы сузить область поиска, *Recoll* позволяет определить ключевые слова, а затем фильтрует их по Все сразу [All Of These], Ни одно [None], Любое из [Any] и другим подобным условиям с помощью усовершенствованного диалогового окна поиска. Можно также ограничить область поиска конкретным типом MIME: например, PDF или электронными таблицами. Наконец, если вы знаете примерное местоположение файла, укажите поиск в конкретных подпапках.

Большинство других инструментов постоянно следят за вашим диском и сохраняют индекс при любых изменениях, а *Recoll* по умолчанию создает только статический индекс. А значит, при желании получать свежие результаты поиска вы должны вручную обновлять индекс, нажав Файл > Обновить. Впрочем, периодическую индексацию сделает за вас *Cron*, при соответствующей настройке. Это и благо, и бремя: системные ресурсы не тратятся на постоянные попытки поддерживать индекс в актуальном состоянии, но по сравнению с другими инструментами требуются лишние действия.

Луч надежды, по крайней мере для желающих скомпилировать *Recoll* самим, дают два инструмента: *File Alteration Monitor (Fam)* и *Inotify*, они следят за файловой системой и ее изменениями. При компиляции *Recoll* можно включить поддержку любого из них с **--with-fam** или **--with- inotify**.

Recoll не создан для индексации всех типов файлов. Для должной работы с PDF, MP3, RTF, MS Office и нескольких других экзотических форматов придется установить дополнительные пакеты, такие как *Antiword* (для *MS Word*) и *Catdoc* (для *MS Excel* и *PowerPoint*). Без этих инструментов индексироваться будут только имена файлов, и *Recoll* не предоставит фрагменты или функцию предпросмотра.



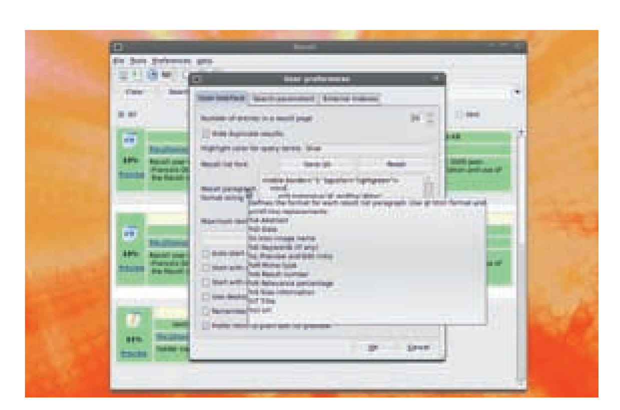
Recoll

Версия: 1.12.1
Сайт: lesbonscomptes.com/recoll
Цена: Бесплатно под GPL

» Находится в жесткой конкуренции с *Beagle*. Возможность создания статических индексов – реальное преимущество.



9/10



› Используя только HTML, можно легко менять вид страницы результатов поиска. Это просто, прикольно и необычно.

Поисковики рабочего стола

Вердикт

Recoll 9/10

Самым поразительным аспектом настольных поисковых инструментов является то, что многие из них еще не достигли версии 1.0. Тем не менее, каждая программа становится все более впечатляющей с каждым релизом. Хотя большинство их функций остаются такими же, почти каждое приложение имеет нечто уникальное.

Аппетит по сжиранию дискового пространства у настольных поисковиков заставит позавидовать даже горных приматов, и для применения их в вашей системе нужно делать выбор с оглядкой. Хотя большинство из этих инструментов отпраздновали уже несколько дней рождений, поиск на рабочем столе в целом (и этими средствами в частности) не имеет большой базы пользователей. Все же найдется больше людей, обсуждающих *grep*, чем *Recoll*.

Быть может, это связано с заблуждением, что настольный инструмент поиска требует много ресурсов. Но ничто не может быть дальше от правды.

Топ-три

Хотя мы не избалованы выбором, было нелегко распределить финальную тройку, потому что три претендента на пьедестал *Google Desktop*, *Beagle* и *Recoll* находятся в жесткой конкуренции. Будем надеяться, что это способствует развитию.

Но сначала рассмотрим поисковики, не попавшие в тройку. Худшее, что можно сказать о *Strigi* – отсутствие навигации по страницам. Зачем нужен инструмент, скрывающий результаты поиска? А вот *Tracker*, однако, имеет все задатки претендента. Возможность указывать, какие каталоги смотреть и индексировать означает меньший размер индекса, поэтому он получает 6/10, несмотря на неполадки с тэгами и медленный запуск.

На последнем месте в тройке финалистов – *Google Desktop*. Эта программа получает бонусные баллы за использование браузера для отображения результатов, но расширенный раздел поиска все еще требует улучшений. Функция автоматической подсказки при поиске полезна, хотя ее ограничение показом только верхних шести результатов в апплете по умолчанию – явный огрех дизайна.

> «Определение ключевых слов помогает Recoll точно найти файл.»

Затем, под номером два, идет несправедливо оболганный *Beagle*. Одна из сильных сторон этого инструмента – малый размер индекса, особенно по сравнению с *Strigi* и *Google Desktop*. При добавлении web-интерфейса будущее *Beagle* видится весьма светлым.

Невзирая на мощную конкуренцию со стороны *Beagle*, особенно по числу поисковых параметров, *Recoll* удается уцепиться за первое место. Заставлять пользователя индексировать каталоги вручную вовсе не глупо. Тому, кто убежден, что настольные поисковики съедают многовато ресурсов, он дает возможность индексировать систему в удобное для него время. И это должно помочь *Recoll* в привлечении пользователей, которые обычно сторонятся подобных инструментов.

Определение ключевых слов для поиска и избегания при поиске помогает *Recoll* всякий раз точно найти местонахождение файла. Другим инструментам не мешает тоже обдумать включение этой функции. LXF



› ***Recoll*** **выполняет предпросмотр почти всех типов файлов, и можно даже искать внутри предпросмотра.**

Обратная связь

Думаете, критика *Beagle* является оправданной, и должен был победить *Google Desktop*, так как он умеет индексировать Gmail? Должны ли инструменты для поиска на рабочем столе учитывать тэги? Согласны ли вы, что *Strigi* плох, или это сугубо наше мнение? Присылайте нам ваши мысли о данном Сравнении по электронной почте на letters@linuxformat.ru.

Таблица характеристик

Название	Версия	Применение шаблонов	Поиск в архивах	Поиск в PDF	Поиск в OGG/MP3	Поиск в почте	Поиск в закладках и истории браузера	Поиск в файлах OOo	Стемминг	Тэги пользователя	Поиск в резервных копиях
Google Desktop	1.2.0	✘	Только по имени файла	✔	✔	✔	Частично	✔	✔	✘	✔
Tracker	0.6.93	✔	Только по имени файла	✔	✔	✔	✘	✔	✔	✔	✘
Strigi	0.6.3	✔	✘[1]	✘[1]	✘[1]	✘	✘	✘[1]	✘	✘	✘
Beagle	0.3.9	Частично	Только по имени файла	✔	✔	✔	✔	✔	✔	✘	✘
Recoll	1.12.1	✔	Только по имени файла	✔	✔	✔	✔	✔	✔	✔	✔

[1] Несмотря на ввод точного имени файла, *Strigi* не нашел ни архива, ни MP3, ни PDF. Поиск *.tar, *.pdf и *.mp3 результатов не дал.

C++ Матрицы и векторы

Семен Есилевский рассматривает четыре библиотеки, реализующих концепции линейной алгебры, и проводит сравнение их производительности.



Наш эксперт

Семен Есилевский
Научный сотрудник, вычислительный программист и энтузиаст Linux и Open Source. Кандидат физико-математических наук, работает в области теоретической биофизики и молекулярного моделирования.

Современный прикладной программист, работающий с C++, может годами не вспоминать о том, что компьютеры были изначально созданы для интенсивных математических вычислений. Однако рано или поздно почти каждому разработчику приходится «возвращаться к истокам» и писать вычислительный код, который должен выполняться максимально быстро.

Практически любой вычислительный алгоритм включает множество действий с векторами и матрицами; иными словами, практически все вычислительное программирование базируется на операциях линейной алгебры над одно- или двумерными массивами. Стандартные массивы C++ плохо приспособлены для этих целей:

- Каждый массив – всего лишь указатель, по которому приходится вручную выделять и освобождать память. У матриц это необходимо делать для каждой строки.
- Массивы «не знают» своего размера. Вместе с ними приходится хранить и передавать соответствующие длины.
- Любые действия с массивом как с целым приходиться расписывать поэлементно в виде циклов.

Если используются только векторы (что бывает крайне редко), то на помощь приходит класс **valarray** из STL. Можно взять и **std::vector**, но у него гораздо меньше возможностей – в частности, нет перегруженных математических функций для поэлементного доступа. Если же требуется работать с матрицами, то волей-неволей приходится использовать специализированные сторонние библиотеки.

В идеале библиотека линейной алгебры для C++ должна поддерживать запись выражений с векторами и матрицами в естественной математической нотации, иметь удобный интерфейс, поддерживать рутинные операции высокого уровня (такие как решение систем линейных уравнений) и обеспечивать максимальную производительность. Также желательно наличие удобного интерфейса к стандартным библиотекам линейной алгебры *BLAS* и *LAPACK*, написанным, как правило, на Fortran (зависит от поставщика) и прекрасно оптимизированным для разных типов процессоров. Производительность этих библиотек считается непревзойденной, но использовать их напрямую в C++ довольно сложно из-за архаичного интерфейса и другого порядка хранения чисел в матрицах.

> «Компьютеры изначально создавались для математики.»

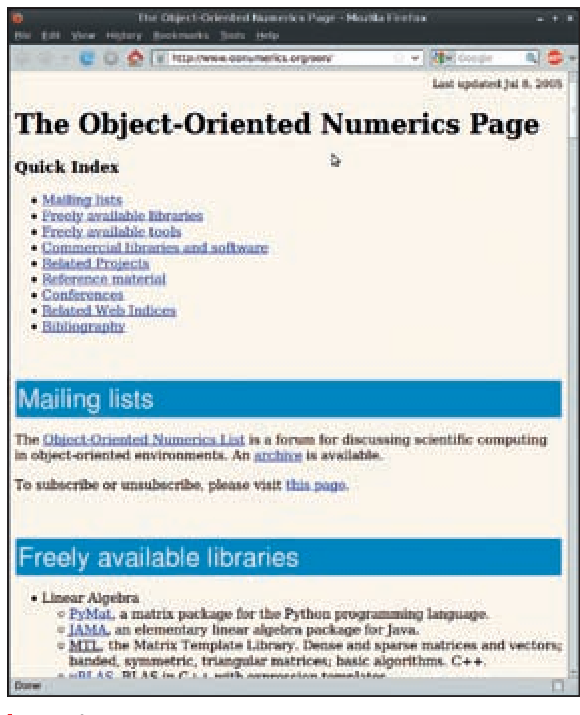


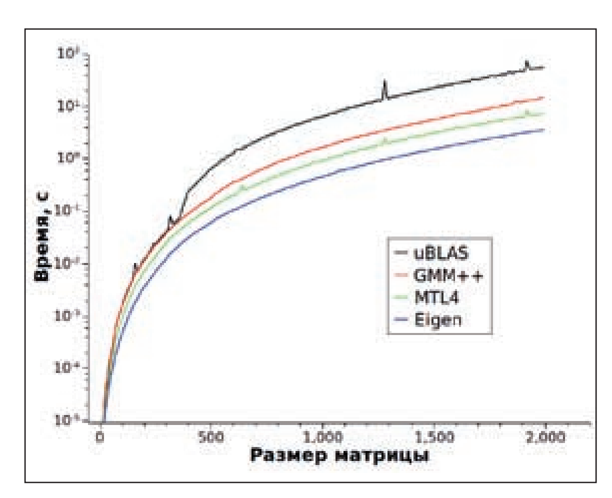
Рис. 1. Oonumerics.org — портал для вычислительных программистов, работающих с объектно-ориентированными языками.

Цена универсальности

Создание класса, инкапсулирующего векторы и матрицы – любимое упражнение в учебниках по C++, позволяющее продемонстрировать множество полезных приемов. Однако такие «доморощенные» классы неизбежно имеют крайне низкую производительность. Проблема состоит в том, что компилятор C++ не обрабатывает векторы и матрицы в выражениях поэлементно, как это необходимо для оптимальной производительности. В результате в сумме матриц

```
mat1 = mat2 + mat3;
```

правая часть сначала будет вычислена и присвоена некоему временному объекту. Затем этот временный объект будет скопирован в mat1 и разрушен. Очевидно, что вызов конструктора и деструк-



› **Рис. 2. Время перемножения матриц разного размера (чем меньше, тем лучше).**

тора временного объекта и двукратное копирование данных совершенно излишни, но такова цена универсальности языка C++. Единственный действенный выход из этой ситуации состоит в использовании шаблонов выражений [expression templates]. Эта методика по праву считается одной из самых сложных в программировании на C++, и подробно описывать ее в рамках данной статьи нет смысла. В двух словах можно сказать, что классы, использующие шаблоны выражений, берут на себя часть работы компилятора. Выражения преобразуются путем рекурсивной подстановки шаблонов на этапе компиляции, так что компилятор генерирует оптимальный код без использования ненужных временных переменных. В хорошо спроектированной библиотеке вся эта сложная работа скрыта за простыми интерфейсами классов.

На практике имеет смысл использовать только те библиотеки, которые применяют шаблоны выражений, так как только они имеют производительность, близкую к низкоуровневому коду с явными циклами.

В поисках идеала

Существует множество библиотек линейной алгебры для C++ (внушительный, хотя и неполный, список доступен на портале www.oonumerics.org). Даже если отбросить те, которые не используют шаблонов выражений и, следовательно, имеют очень низкую производительность, остается более десяти вариантов. Ниже приведены краткие характеристики четырех наиболее мощных и удобных (с точки зрения автора) библиотек. Для каждой из них представлен фрагмент кода, позволяющий решить систему линейных уравнений и демонстрирующий основные особенности синтаксиса.

» uBLAS

Библиотека *uBLAS* [1] входит в широко используемый набор библиотек *Boost*, что обеспечивает ей популярность и позволяет надеяться на стандартизацию в будущем. *uBLAS* поддерживает все основные операции линейной алгебры с плотными и разреженными матрицами разного типа. Имеются функции для решения систем линейных уравнений (впрочем, не очень удобные). Для *uBLAS* существует хороший интерфейс к *LAPACK*, но его необходимо устанавливать отдельно, так как он не входит в стандартный набор библиотек *Boost*. Официальная документация *uBLAS* откровенно разочаровывает. Она переполнена техническими деталями и не содержит даже простейших примеров рутинных операций.

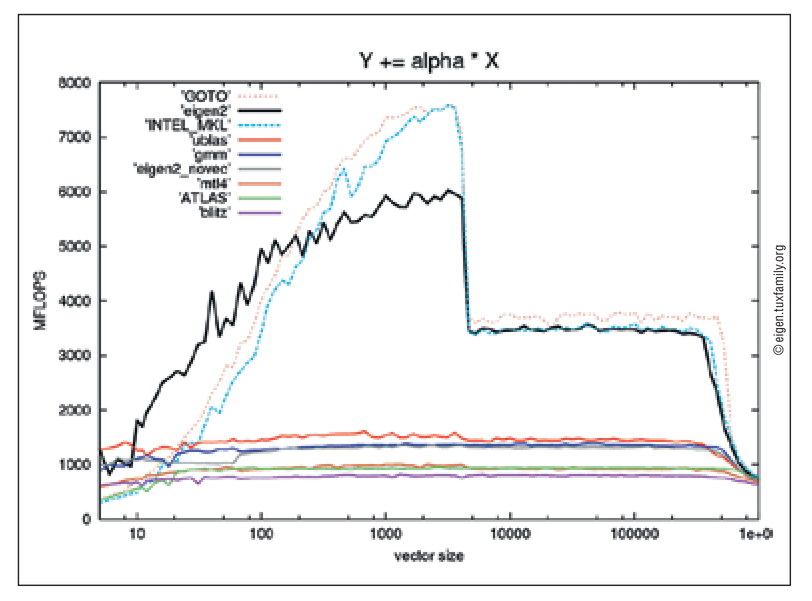
```
// Заголовочные файлы входят в общее дерево заголовков
boost
#include <boost/numeric/ublas/matrix.hpp>
#include <boost/numeric/ublas/vector.hpp>
// Заголовок для решения систем линейных уравнений сила-
ми uBLAS
#include <boost/numeric/ublas/lu.hpp>
// Интерфейс к LAPACK (нужно устанавливать отдельно)
#include <boost/numeric/bindings/traits/ublas_matrix.hpp>
#include <boost/numeric/bindings/lapack/gesv.hpp>
...
// Пространства имен весьма громоздкие, лучше определить
синонимы
namespace lapack = boost::numeric::bindings::lapack;
namespace ublas = boost::numeric::ublas;
// Для совместимости с LAPACK объявляем матрицу упорядо-
ченной по колонкам
ublas::matrix<double, ublas::column_major> A(N,N);
ublas::vector<double> b(N), x(N);
...
// Решаем систему уравнений силами uBLAS
// (довольно громоздко и довольно медленно...).
ublas::permutation_matrix<std::size_t> pm(A.size1()); // Матри-
ца перестановок
ublas::lu_factorize(A, pm); // Факторизуем матрицу
ublas::lu_substitute(A, pm,x); // Ищем решение
// Решаем систему уравнений силами LAPACK (намного
быстрее).
//Правая часть должна быть матрицей
ublas::matrix<double, ublas::column_major> temp(N,1);
column(temp,0) = b; // Доступ к колонкам матрицы реализован
процедурами
lapack::gesv(A,temp); // Вызов функции LAPACK
x = column(temp,0); // Получаем вектор решения
// Проверяем точность решения
b = prod(A,x);
// Произведение матрицы на вектор не записывается в естест-
венной нотации A*x
// (как и большинство других операций)
```

» MTL4

Библиотека *MTL4* (*Matrix Template Library*), доступная по адресу [2], как видно из ее названия, изначально построена на шаблонах выражений и имеет очень хорошую производительность в сочетании с естественной математической нотацией. Особое внимание уделено эффективности хранения и обработки разреженных матриц. К сожалению, стремление к технически совершенным способам обработки матриц разного типа выливается в усложнение тривиальных операций. Например, извлечение столбца из матрицы реализовано крайне громоздко и неинтуитивно – с помощью итераторов и курсоров. Документация *MTL4* на первый взгляд достаточно подробная и понятная, но при ближайшем рассмотрении оказывается, что многие примеры не компилируются, а использованные в них типы данных и функции попросту не определены.

```
// Несмотря на путь к заголовочным файлам, MTL4 не входит
в boost
#include <boost/numeric/mtl/mtl.hpp>
// Пространство имен простое и удобное
using namespace mtl;
//Плотная матрица NxN
dense2D<double> A(N,N);
// Плотные векторы
dense_vector<double> x(N), b(N);
…
// Линейная система решается предельно интуитивно
x = lu_solve(A,b);
// Проверяем точность решения
b = A*x; // Естественная математическая нотация
```

»

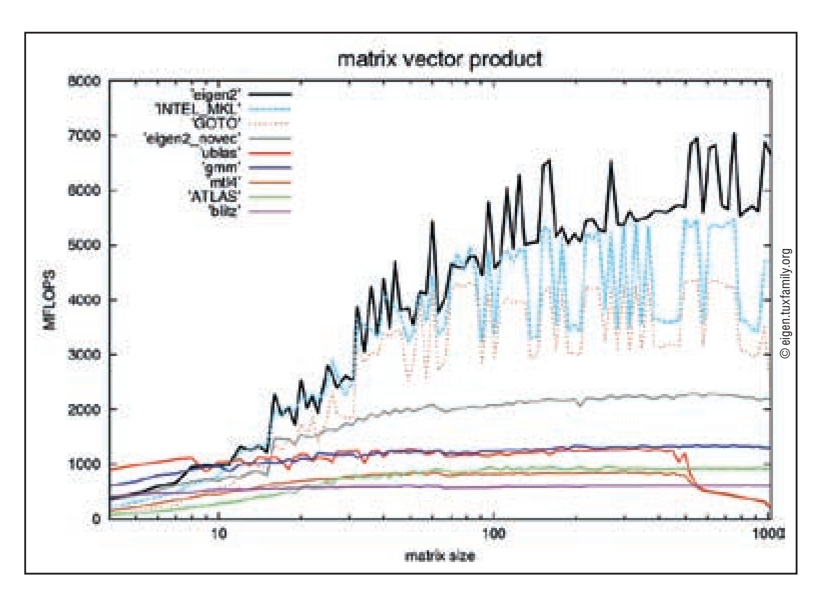


› Рис. 3. Скорость умножения вектора на скаляр в зависимости от размера вектора для разных библиотек.

» GMM++

GMM++ [3] является составной частью мощной и развитой библиотеки *Getfem++* для работы с методами конечных элементов (в том числе для решения систем дифференциальных уравнений). Дизайн *GMM++* «навеян» библиотекой *MTL* ранних версий. Поддерживаются плотные и разреженные матрицы разного типа, которые можно сочетать в выражениях произвольным образом. Имеются все основные стандартные алгоритмы (решение систем линейных уравнений, нахождение собственных значений) и удобный интерфейс для *LAPACK* и популярной библиотеки для разреженных матриц *SuperLU*. *GMM++* демонстрирует очень хорошую производительность; документация подробная и понятная. К недостаткам *GMM++* относятся полное отсутствие естественной математической нотации (интерфейс сугубо процедурный), неунифицированный доступ к элементам векторов и матриц (через **operator[]** и **operator()** соответственно), отсутствие специализированных типов для массивов малого размера.

```
// Интерфейс с LAPACK включается глобальной директивой
#define GMM_USES_LAPACK
// Основной заголовок
#include <gmm/gmm.h>
gmm::dense_matrix<double> A(N,N);
// В качестве класса для плотных векторов используется
std::vector,
// и никаких дополнительных «удобств» он не дает
std::vector<double> x(N), b(N);
// Доступ к элементам матрицы и вектора не унифицирован
(операторы () и [])
A(2,67) = 1.0;
b[5] = 2.0; // b(5), вопреки ожиданиям, не работает
...
// Линейная система решается просто
gmm::lu_solve(A, x, b);
// Все операции записываются в процедурном виде,
// естественная нотация не поддерживается
mult(A,x,b);
// Доступ к срезам матриц довольно громоздкий
gmm::sub_matrix(A, gmm::sub_interval(2, 3), gmm::sub_
interval(2, 3));
// Доступ к колонкам и строкам матриц прост, но нотация
// процедурная, а не объектно-ориентированная
gmm::mat_row(A, 5);
```



› Рис. 4. Скорость умножения матрицы на вектор в зависимости от размера матрицы для разных библиотек.

» Eigen

Eigen [4] является официальной математической библиотекой для среды KDE, но не имеет никаких внешних зависимостей, в том числе и от KDE. Особое внимание здесь уделено элегантности и интуитивности интерфейса. *Eigen* использует автоматическую параллелизацию многих операций для процессоров, поддерживающих инструкции SSE2. Для часто используемых векторов и матриц размера 2, 3 и 4 используются специальные типы данных, что резко повышает производительность. Сообщения об ошибках в *Eigen* вполне читабельны, что является большой редкостью для библиотек, использующих шаблоны. Многочисленные внутренние проверки в режиме отладки снижают количество «глупых» ошибок. Документация заслуживает всяческих похвал и содержит в сжатом виде всю информацию, необходимую для применения библиотеки, без лишних технических подробностей.

Из недостатков *Eigen* можно назвать некоторые нюансы использования, вызванные встроенной параллелизацией, и отсутствие интерфейса к *LAPACK*. Некоторые функции пока не реализованы, так как библиотека интенсивно развивается.

```
// Общий заголовок
#include <Eigen/Core>
// Заголовок для решения систем уравнений
#include <Eigen/LU>
// Макрос для импорта основных типов данных и функций в текущее пространство имен
USING_PART_OF_NAMESPACE_EIGEN;
// Динамическая матрица типа double
MatrixXd A(N,N);
// Динамический вектор типа double
VectorXd b(N),x;
// Матрица фиксированного размера 3x3
// работает намного быстрее чем MatrixXd s(3);
Matrix3d s;
// Решаем линейную систему. Нотация полностью объектно-
ориентированная.
A.lu().solve(b, &x);
// Проверяем точность решения
// Естественная математическая нотация
b = A*x;
// Доступ к строкам, столбцам и срезам очень прост
b = A.col(k);
A.row(k) = x;
//Заполнить нулями 10 элементов строки k начиная с третьего
A.row(k).segment(3,10).fill(0.0);
```

Тест производительности

Производительность рассмотренных библиотек оценивалась в четырех тестах, соответствующих разным моделям использования. В первом тесте 10 раз решалась система из 2000 линейных уравнений. Матрица коэффициентов и правая часть заполнялись случайными числами. В случае *uBLAS* и *GMM++* система уравнений решалась как с помощью интерфейса к *LAPACK*, так и силами самой библиотеки. Во втором тесте 100 раз производилось умножение случайной матрицы размера 2000×2000 на соответствующий вектор. Третий тест был аналогичен второму, но использовалась матрица размера 3×3 и 1 000 000 умножений. В третьем тесте для *uBLAS* и *Eigen* использовались как обычные типы данных, так и специализированные для маленьких массивов фиксированного размера. В четвертом тесте оценивалось время перемножения двух случайных матриц, размер которых варьировался от 10 до 2000.

Тест проводился на компьютере с процессором Intel Core 2 Quad Q9300 (тактовая частота 2.5 ГГц) с 32-битной системой Ubuntu Linux 9.10. Использовался компилятор *gcc 4.4.1* с флагами оптимизации '**-msse2 -O3 -march=native -DNDEBUG**'. В случае *Eigen* использовалась оптимизация **-O2**, так как более агрессивная оптимизация снижала производительность. Применялась стандартная открытая реализация библиотеки *LAPACK* версии 3.1, входящая в Ubuntu 9.10. Результаты приведены в таблице и на рис. 2 (время в секундах).

В решении системы уравнений без использования *LAPACK* лидерами с очень близкими результатами являются *GMM++* и *uBLAS*, а *MTL4* и *Eigen* отстают примерно в полтора раза. Если использовать *uBLAS* и *GMM++* как интерфейсы для *LAPACK*, то можно добиться более чем двукратного ускорения. В умножении большой матрицы на вектор *Eigen*, *MTL4* и *GMM++* демонстрируют одинаковую производительность, а *uBLAS* отстает почти в 5 раз. При работе с матрицами и векторами размера 3, которые характерны для трехмерной графики, физического моделирования и тому подобных задач, *Eigen* демонстрирует колоссальное преимущество. Даже при использовании обычных типов данных, не оптимизированных для матриц малого размера, *Eigen* справляется со своей работой в 12 раз быстрее конкурентов. При использовании специализированных типов данных это преимущество достигает примерно 20000 раз (!). Для сравнения, специальные типы данных в *uBLAS* улучшают производительность примерно в три раза. Хуже всего справляется с маленькими матрицами *MTL4.*

При перемножении матриц произвольного размера также с существенным отрывом выигрывает *Eigen.* Второе место у *MTL4.* Далее идут *GMM++* и *uBLAS*, которые для небольших матриц демонстрируют одинаковые результаты. Для матриц, больших чем 360×360, производительность *uBLAS* резко падает.

Нужно заметить, что в *GMM++* подключение интерфейса к *LAPACK* происходит «глобально» и оказывает влияние на все операции, в том числе и на те, где дополнительный вызов библиотечной функции «дороже», чем сама операция. В результате при использовании *LAPACK* резко падает производительность для маленьких матриц.

Для *Eigen* оптимизация, более агрессивная, чем **-O2**, снижает производительность. По-видимому, при этом компилятор и внутренние оптимизации библиотеки на уровне шаблонов выражений начинают конфликтовать.

Для *uBLAS* и *GMM++* критичным является использование флага **-DNDEBUG**, отключающего внутренние проверки и улучшающего производительность на порядок. В *Eigen* внутренние проверки отключаются автоматически при уровне оптимизации, большем чем **-O0**.

Рассмотренные здесь тесты демонстрируют производительность библиотек в типичных вариантах использования, но, конечно, не являются исчерпывающими. На рисунках приведены некоторые результаты нескольких более строгих сравнительных тестов с официального сайта *Eigen*.

За кадром

Из библиотек, не попавших в эту короткую статью, можно назвать следующие:

- *Blitz++* (http://www.oonumerics.org/blitz/) – потенциально очень мощная библиотека, однако в ней нет реализации стандартных алгоритмов вроде решения систем линейных уравнений.
- *POOMA* (http://acts.nersc.gov/pooma/) – библиотека, ориентированная на параллельное решение систем дифференциальных уравнений.
- *TNT* и *JAMA/C++* (http://math.nist.gov/tnt/index.html) – «библиотеки-обертки» над *BLAS* и *LAPACK* (наследники *LAPACK++*).
- *SparceLib++* (http://math.nist.gov/sparselib++/) – специализированная библиотека для разреженных матриц.

Выводы

Из четырех рассмотренных нами библиотек наилучшее впечатление производит *Eigen.* Она обеспечивает прекрасную производительность (особенно в работе с маленькими матрицами), поддерживает естественную математическую нотацию, имеет простой и интуитивно понятный интерфейс и очень хорошо документирована.

В большинстве случаев *Eigen* будет оптимальным выбором. Однако поддержка разреженных матриц в *Eigen* все еще считается экспериментальной, поэтому при интенсивной работе с такими матрицами в качестве альтернативы можно рассматривать *MTL4.* Функции для решения линейных уравнений, нахождения собственных значений и аналогичные операции в *Eigen* и *MTL4* работают раза в 1,5–2 медленнее, чем соответствующие функции в *GMM++* и *uBLAS*, а использование в последних интерфейса к *LAPACK* увеличивает этот разрыв еще вдвое. Если данная разница критична (например, для очень больших матриц) и перевешивает неудобства от неестественного процедурного интерфейса, то разумным выбором могут оказаться *uBLAS* и *GMM++* . LXF

Ссылки

1 uBLAS: http://www.boost.org/doc/libs/1_40_0/libs/numeric/ublas/doc/overview.htm
2 MTL4: http://www.osl.iu.edu/research/mtl/mtl4/
3 GMM++: http://home.gna.org/getfem/gmm_intro
4 Eigen: http://eigen.tuxfamily.org/

Библиотека	Система уравнений	Умножение матрицы 2000×2000 на вектор	Умножение матрицы 3×3 на вектор
uBLAS	9.31	**2.84**	**7.41** (matrix, vector) **2.19** (bounded_matrix, bounded_vector)
uBLAS с *LAPACK*	3.81		
MTL4	14.05	0.59	21.37
GMM++	9.2	0.59	3.25
GMM++ с *LAPACK*	3.85	0.6	7.92
Eigen	**14.7**	**0.59**	**0.18** (MatrixXd, VectorXd) <**0.0001** (Matrix3d, Vector3d)



Бальзам для глаз...

Грэм Моррисон бьется с Windows 7 и Mac OS X за самый миловидный рабочий стол. И вам такого же!

Приятная внешность никогда не считалась приоритетом для Linux-приложений. Казалось, давно ли мы боролись за элементарный внешний вид рабочего стола — например, сглаженные шрифты? Однако теперь ситуация изменилась до неузнаваемости.

Сейчас можно смело сказать, что Linux – лидер визуальных эффектов, сущий рог изобилия красивых штучек, наводняющих ваш экран. А при некоторой подстройке он станет выглядеть еще лучше. Поскольку Windows и OS X постоянно поднимают планку ожиданий от рабочего стола для среднего пользователя, чрезвычайно важно, чтобы Linux был впереди – даже если это означает просто включение эффекта джинна для свернутых приложений, когда к вам в гости зашли друзья, или использование более практичной версии виртуальных рабочих столов, если вы даете кому-нибудь попользоваться своим компьютером. Добавление красот вовсе не должно состоять лишь в чисто косметических изменениях. Напротив, оно должно повысить удобство использования и сделать работу более производительной и целеустремленной. Мы просто покажем, как это делается, а по ходу изложения объясним, как украсить ваш рабочий стол Linux, на зависть друзьям – сторонникам проприетарных ОС.

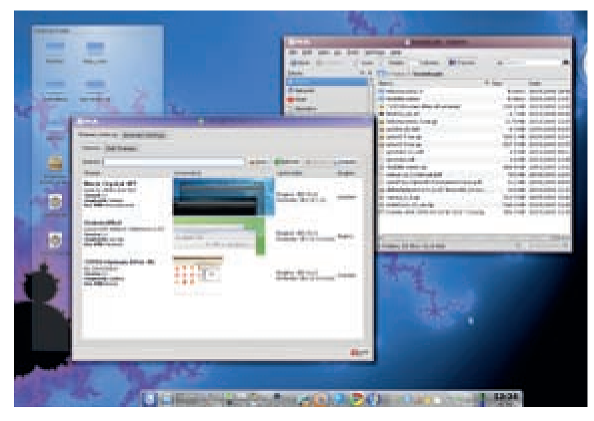


Compiz и Emerald

Используйте лучший декоратор окон в Linux.

Появление проекта *Compiz* стало поворотным пунктом для настольного Linux, однако за несколько лет он испытал немало переходных периодов и неприятностей. Пережив ответвление, отчуждение сообщества и нехватку разработчиков, он возродился как стандартное графическое дополнение к Gnome и многим другим рабочим столам в Linux. Его технология отныне сочетает завершенность с солидным повышением производительности вашего труда. Если у вас он еще не установлен, многие дистрибутивы предлагают готовые пакеты; чтобы использовать *Compiz* во всей красе, вам потребуется лишь 3D-ускоритель средней мощности, поддерживаемый Linux.

Одним из лучших дополнений к *Compiz* является менеджер окон *Emerald*. Он заменяет рамку вокруг окон ваших приложений, и его огромное преимущество по сравнению с предоставляемым по умолчанию — отсутствие привязки к определенному GUI, например, Gnome, KDE или *Xfce*. Также это самый настраиваемый менеджер окон среди всех, что нам встречались, да и выглядит он лучше всех.



› ***Emerald* заменит рамки ваших окон на приятную, легко настраиваемую альтернативу.**

Запуск Emerald

Если *Compiz* у вас уже установлен, *Emerald* должен быть доступен в виде одного добавочного пакета. Если вы используете Gnome с рядом дополнительных эффектов, значит, *Compiz* уже работает, и вы можете запустить менеджер окон *Emerald*, введя **emerald** в терминале. Тем, кто пользуется KDE или еще не установил *Compiz*, надо сначала запустить **decorator**, а затем ввести команду **compiz.real --replace**. Вы увидите, что рамки окон сменятся текущей темой *Emerald*, которую можно отредактировать, запустив программу Emerald Theme Manager, обычно расположенную в меню приложений в разделе Настройки.

По умолчанию Emerald Theme Manager содержит только одну тему – ту, что поменяла рамки окон вашего рабочего стола. Если хотите извлечь из *Emerald* больше, обзаведитесь новыми темами. Лучший способ сделать это – зайти на www.compiz-themes.org, найти темы на свой вкус и скачать их. Обычно они заархивированы в файле **.tar.gz**, и их придется распаковать в какой-нибудь каталог. Затем укажите в менеджере тем *Emerald* файл **.emerald** из этой директории, чтобы импортировать его. Некоторые темы состоят из одного файла **.emerald** и не требуют особой возни.

Настройка ваших тем

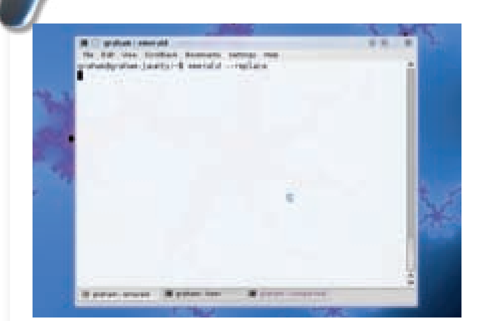
Тонкая настройка темы производится на странице редактирования в менеджере тем. Щелкните по вкладке Изменить темы [Edit Themes], и вы увидите основные для каждой темы параметры, разделенные по пяти категориям. Наиболее влиятельным параметром является Движок окна [Frame Engine], и его можно поменять в выпадающем списке. Как и для остальных параметров *Emerald*, каждое изменение обновляет изображение в реальном времени, и вы сразу видите последствия сделанных вами изменений. Наш любимый эффект называется 'trueglass', но 'oxygen' и 'vrunner' тоже хороши. Каждый движок имеет собственный набор параметров, которые в основном меняют цвета, используемые движком для рамок окон. Из раздела Окно/Тени [Frame/Shadows] можно подстроить размер рамки окна и задать длину тени, а на вкладке Заголовок [Titlebar] настраивается размер рамки, где расположен текст, а также тип подсветки и тени для его отображения. Не забудьте сохранить свою тему, когда закончите.

Скорая помощь

Перемещаться между виртуальными рабочими столами можно прокруткой колеса мыши над рабочим столом.

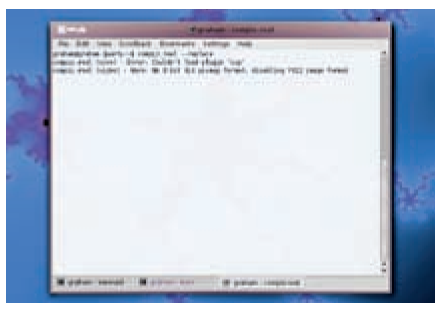


Шаг за шагом: Запускаем Emerald



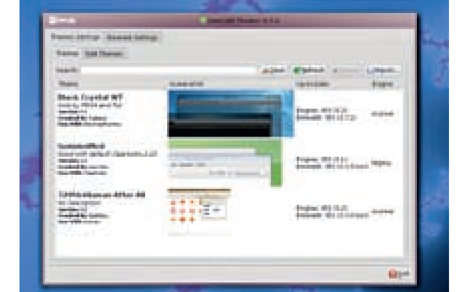
1 Простой способ

Когда *Compiz* и *Emerald* установлены, откройте терминал и введите **emerald --replace**. Запустится менеджер окон *Emerald*.



2 Более сложный способ

Если у вас уже выполняется *Compiz*, первый шаг вам можно не делать. Вместо этого введите **compiz.real --replace** в другом сеансе терминала.



3 И все готово!

Вы увидите смену оформления окон на вашем рабочем столе. Для изменения оформления запустите *Emerald Theme Manager* из меню приложений.



KDE

Возможно, самый гибкий рабочий стол в мире.

Разработка Windows и OS X постепенно повысила ожидания людей относительно того, как должно выглядеть и действовать окружение рабочего стола, и этот момент не был упущен создателями KDE. Это единственный стандартный рабочий стол, содержащий бесплатные эффекты и украшательства, которые можно включить несколькими щелчками мыши.

Согласно философии KDE, он также позволяет глубоко вникнуть в механизм создания тем. Можно поменять почти все, что касается прорисовки отдельных компонентов – от типов значков и эффектов для них до границы окна и отрисовки полос прокрутки. Экспериментировать с KDE можно сутками напролет. Однако лучше начать с эффектов.

Скорая помощь

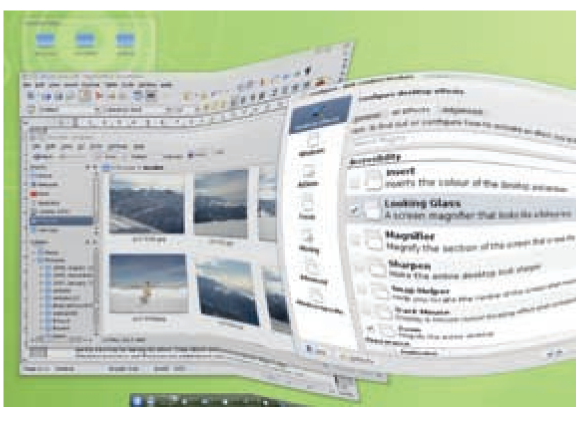
Если при применении эффекта иконка не меняется, то похоже, что она использует несовместимое изображение. Щелкните правой кнопкой по иконке и выберите Параметры иконки [Icon Settings]. В появившемся окне щелкните по самому значку и выберите один из тех, что предлагаются KDE в качестве замены.

Эффекты

В KDE 4.3 включен обширный набор эффектов, улучшающих и внешний вид, и вашу продуктивность. Большинство из них требуют видеокарты с 3D-ускорением. Она не обязана быть мощной, и сойдет любой адаптер от Intel, Nividia и ATI/AMD, если у вас стоят не слишком старые драйверы. Все эффекты KDE спрятаны под пунктом меню Настроить поведение окна [Configure Window Behaviour], которое возникает при щелчке правой кнопкой мыши на заголовке последнего. В первом окне включаются композитные эффекты и выбираются эффекты перехода при смене окон и рабочих столов. Именно здесь находится пресловутый куб; но наш любимый эффект для переключения между рабочими столами называется Slide. Он эмулирует поведение Spaces из OS X, плавно передвигая окна из одного виртуального рабочего стола на другой. Slide не так радует глаз, зато он логичнее, и, что более важно, практичнее.

«В KDE 4.3 огромный диапазон эффектов, улучшающих работу.»

KDE есть KDE; и, переключившись на вкладку Все эффекты [All Effects] страницы Эффекты рабочего стола [Desktop Effects], вы увидите гораздо больше эффектов. Например, нажав на значок гаечного ключа справа от Desktop Cube, можно подстроить масштаб и прозрачность, а также указать горячую клавишу для



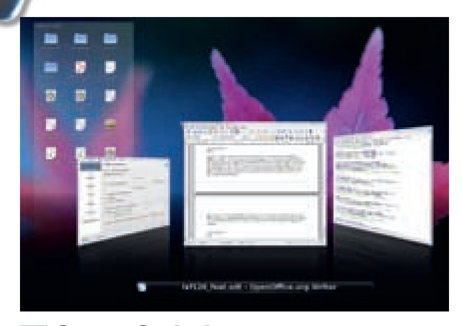
› Эффект лупы, может, и не столь полезен, как реальная лупа, но смотрится замечательно.

запуска эффекта. Или, допустим, KDE решил рисовать тени рабочего стола синим цветом; это исправляется перенастройкой эффекта тени – сменой цвета на черный, а прозрачности, нечеткости и размера – на значения около 10.

Другие стоящие эффекты – Magic Lamp, Minimise Animation и Explosion для пущего шика при сворачивании или закрытии окон; нам также по душе эффект Wobbly Window [пульсирующее окно], если слайдер wobbliness установить на Less. Опция Sheet будет складывать диалоги открытия файлов в точку и из точки исчезновения; еще мы включили эффект Dim Inactive, который приглушает цвет неактивных окон на вашем рабочем столе и подсвечивает приложение, активное в данный момент. Мы установили силу этого эффекта равной 10 и поставили опцию Apply Effect To Groups.

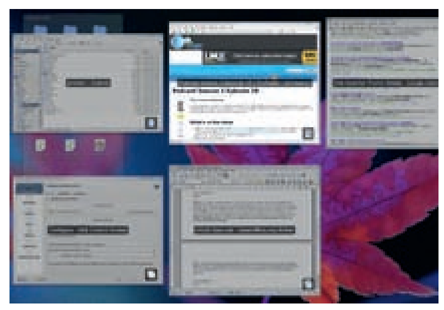
После эффектов, самое большое изменение на рабочем столе вызовет смена используемых значков. Для своих приложений KDE применяет стандартные наборы иконок; то есть, поменяв

Три классных эффекта переключения окон



1 Cover Switch

Хорошая замена для стандартного переключения окон, с добавочными украшательствами.



2 Present Windows

Это лучшее решение, если у вас всегда открыто много окон.

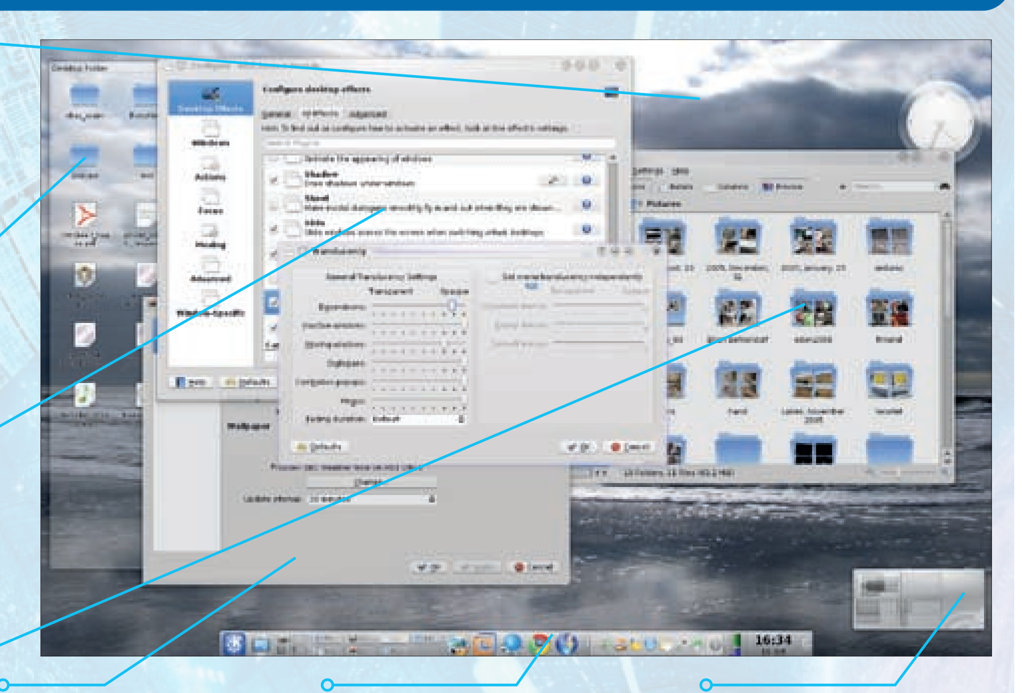


3 Flip Switch

Тот же Cover Switch, но лучше обрабатывает большое количество окон.

Идеальный KDE

Извлечем максимум из своенравного окружения



Динамический фон
В KDE 4.3.1 можно выбирать рисунок, представляющий погоду, компьютерный вирус, трехмерный глобус или интерактивный визуализатор фракталов Мандельброта.

Просмотр папки
Этот плазмоид не только для локальных каталогов: вы можете указать ему сетевые диски и удаленные серверы.

Эффекты
Щелкните правой кнопкой по границе окна и выберите Настроить поведение, а затем Все эффекты. Там будут перечислены все эффекты KDE, установленные в вашей системе.

Миниатюры рисунков
Сейчас KDE может отображать предпросмотр любых изображений, содержащихся в папке. Он включается выбором пункта Preview в меню просмотра.

Полупрозрачность
Сделайте панель Декорации [Decorations] чуть просвечивающей, чтобы добиться эффекта полупрозрачности ваших окон.

Масштабирование панели
Панель можно менять в размере и центровке, чтобы улучшить внешний вид. Она столь же хорошо работает, будучи прикрепленной вертикально к правому или левому краю экрана.

Пейджер рабочего стола
Добавьте плазмоид пейджера на фон рабочего стола, чтобы можно было легко переключаться между ними.

значок один раз, вы увидите результат во всех программах, совместимых с вашим рабочим столом. Значки для перемещения вперед, назад, файловых диалогов, списков, предпросмотров, шрифтов и приложений поменяются все разом, если вы установите новый набор иконок.

Новые значки можно скачать, установить и активировать в разделе Иконки [Icons] – он находится в панели Внешний вид [Appearance] программы Параметры системы [System Settings]. Однако нам не удалось заставить заработать автоматическую установку. Вместо этого мы просмотрели библиотеку иконок на www.kde-look.org, скачали понравившиеся в виде файлов **tar.gz** и с помощью диалога, проявляющегося, если щелкнуть Установить тему из файла [Install Theme File], указали расположение загруженного архива. Для успешного завершения процесса необходимо, чтобы в пакете был конфигурационный файл темы KDE, но если он присутствует, то в списке тем значков на панели Иконки вы увидите новую запись. Просто выберите ее и нажмите Применить [Apply], чтобы изменения возымели силу.

Цвета

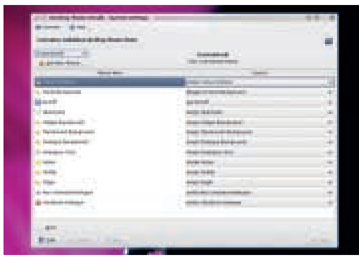
Цвета играют важную роль, и жалко, что в основном они довольно статичны. Большинство дистрибутивов и рабочих столов предпочитают безопасный консерватизм, оставляя пользователю шанс поиграть в Вивьен Вествуд. Это отчасти объясняет, почему в KDE вот уже много лет все остается синим: разработчики, похоже, к нему неравнодушны, и никто не стремится выдумать что-нибудь получше. Такая же инертность наблюдается в неком коричневом дистрибутиве...

Конечно, в отличие от большей части рабочего стола, палитра, применяемая для раскраски вашего экрана, довольно субъективна – как, скажем, цвет потолка в вашей спальне. Возможно, тут разработчики и правы, но почему бы не исследовать другие варианты? Различные цветовые опции можно найти на вечно живой странице Внешний вид; переключитесь на вкладку Цвета [Colours], чтобы увидеть список компонентов, обладающих определенным оттенком. Добраться до нужной части UI удастся методом проб и ошибок, а задание множества цветов за один раз напоминает просмотр каталога красителей; но усилия окупаются. Найдя подходящий тон, который вы захотите скопировать на другие виджеты, нажмите на кнопку Добавить к избранным цветам [Add To Custom Colours], и теперь повторный выбор того же цвета станет простой операцией наведи-и-щелкни.

Детали темы рабочего стола

Панель настроек Параметры темы [Desktop Theme Details] стала частью KDE начиная с версии 4.2, и лучше всего подходит под название «миксер тем». Вы можете включать и выключать различные компоненты или создать отдельную тему KDE. Например, можно взять меню запуска из одной темы, фоновую панель из другой, а стиль строки-подсказки – из третьей, и сохранить все это как свою собственную пользовательскую тему.

Слева от окна перечислены различные компоненты рабочего стола KDE, а справа от каждого из них есть выпадающий список установленных тем. Если нужно поменять тему для меню запуска, просто выберите справа от иконки «Kickoff» ту тему, которую вы хотите использовать. За парой исключений, компоненты интуитивно понятны. В том же окне можно скачать и установить другие онлайновые темы, и переключаться между уже созданными предустановками. Решив подарить свое творение миру, нажмите на маленькую кнопку Еще [More] в левом нижнем углу экрана, и вы сможете дать своей теме имя, добавить благодарности и экспортировать конфигурацию в файл.



› Окно Desktop Theme Details — прекрасный способ управлять темами KDE.



Gnome

Это только на первый взгляд кажется, что в нем нет никаких настроек.

Gnome – блестящий пример минималистского дизайна. За последние несколько лет его интерфейс стал намного более грамотным и доходчивым. Просто взгляните на страницу Визуальные эффекты [Visual Effects] панели Внешний вид [Appearance Preferences], чтобы понять это. В отличие от KDE, чья страница эффектов, кажется, не имеет конца, в Gnome есть только три опции – Нет [None], Обычные [Normal] и Экстра [Extra], и именно здесь делаются самые важные изменения.

Например, для смены цвета фона панели на сочетающийся с рабочим столом выберите соответствующий оттенок и подвигайте ползунок непрозрачности до достижения желаемого результата. В итоге эффект станет намного приятнее, чем стандартная заливка серым, и вы можете сделать это на любой панели вашего рабочего стола, даже на той, что по умолчанию находится на нижней границе – хотя она выглядит не так красиво, когда появляются заголовки окон, поскольку они не перенимают новую тему.

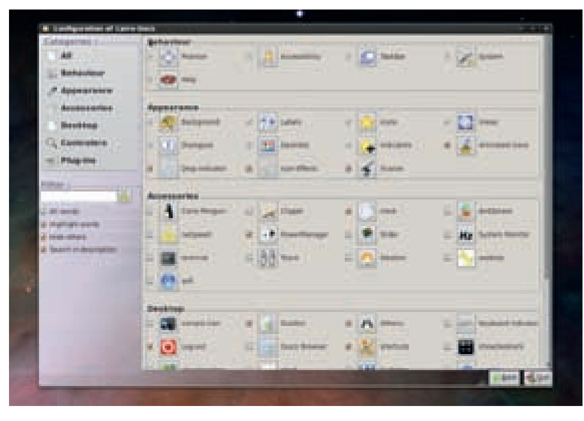
Дистрибутив с особенно удачным рабочим столом Gnome – Linux Mint. Здесь предприняли два отважных шага: убрали с экрана верхнюю панель и резко сменили всю цветовую палитру. С помощью Gnome'овской панели Внешний вид вы можете сами посмотреть, что изменилось. Все параметры настройки аккуратно интегрированы с рабочим столом Gnome и собирают все, что вы видите, в тему. Список тем можно увидеть при первом открытии окна, и, щелкнув по другой теме, можно изменить все, от типа шрифтов и значков, используемых на рабочем столе, до фонового рисунка и цветовой палитры.

Как и в KDE, здесь есть множество эффектов рабочего стола на выбор, и если KDE выигрывает в количестве предлагаемых эффектов, Gnome имеет преимущество в более гладкой интеграции с *Compiz*. Однако для того, чтобы получить еще больше, вам следует установить пакет **ccsm**. Его название – аббревиатура от *Simple Compiz Config Settings Manager*. После инсталляции вы сможете выбрать дополнительный пункт на странице Визуальные эффекты > Специальные [Visual Effects > Custom].

> «Cairo-Dock, несомненно, самая лучшая замена панели Gnome.»

Скорая помощь

Если вы пользуетесь Google Mail в браузере, вы можете поменять цвета темы, используемой для отображения вашей почты, щелкнув по Настройки, затем Темы, и наконец, Выбрать ваши собственные цвета.



› У *Cairo-Dock*, возможно, самая исчерпывающая панель настройки из известных нам приложений.

С блеском — запросто

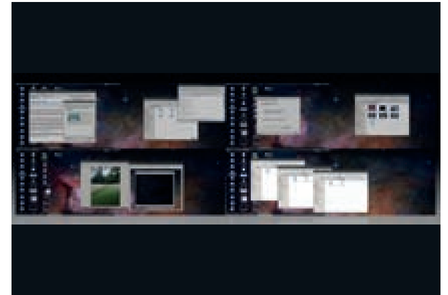
Simple Compiz Config Settings Manager очень похож на список Все эффекты в KDE. Например, перейдя на вкладку Эффекты [Effects], можно сменить тип переключения между приложениями: поддерживаются все варианты, имеющиеся у Конкурента. На вкладке Рабочий стол [Desktop] вы увидите меню, где можно выбрать куб рабочего стола, хотя по умолчанию стоит Стенка [Desktop Wall]. При выборе Cube на странице Эффектов появятся соответствующие опции, а в выпадающем списке Деформация [Deformation] можно свернуть куб в цилиндр или сферу. Нажмите Alt и курсорную клавишу влево или вправо, чтобы увидеть эффект в действии. Если у вас два экрана, вы увидите два куба. Другая умная добавка – эффект масштабирования экрана (Screen Zoom), он находится на вкладке Специальные возможности [Accessibility]. Включив его, удерживайте клавишу Windows на клавиатуре и прокрутите колесо мыши, чтобы экран плавно изменил размер вокруг ваше-

Три трехмерных стола



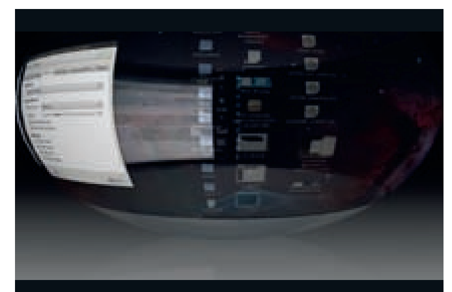
1 Цилиндр

Немного более интуитивный, чем Куб, и не так раздражает. Эффект цилиндра наполовину красив, наполовину практичен.



2 Стенка

Это самый удобный эффект виртуального рабочего стола, и, видимо, поэтому он выбран в Apple и Gnome по умолчанию.



3 Сфера

Идеален по красоте, но нас не покидает мысль, что сфера должна быть чуть ниже. И мы бы добавили немного резвости.

Идеальный рабочий стол Gnome

Исчезающая панель
Мы заменили нижнюю панель на последний релиз *Cairo-Dock*, а панель вверху оставили для управления задачами и апплетов.

Новые иконки
Откройте страницу Внешний вид и нажмите на Установить новую тему [Install New Theme], чтобы добавить новый каталог иконок в разделе Своя тема [Custom Theme]. Эта коллекция называется CON-GFlatSVG.

Новые цвета
Мы скомбинировали оформление окон Dust со своим собственным набором цветов.

Фоновые рисунки
Нам нравится брать свежие изображения в архиве Astronomy Picture of the Day (http://apod.nasa.gov/apod).

Виджеты
Очень популярны легко устанавливаемые *gDesklets*, а также Desktop Gadgets от Google.

Информативные иконки
Cairo-Dock может отмечать полезную информацию на пиктограмме приложения.

Cairo-Dock
Не все темы *Cairo-Dock* столь выдающиеся, как эта, но функциональность у всех на высоте.

го курсора. Это очень удобно, если у вас есть сложности с чтением текста с экрана из-за зрения, но также пригодится, когда вам захочется увеличить изображение или видео.

Прежде чем закрыть окно *CCSM*, взгляните на последнюю страницу. Там представлена панель настройки, задающая действия, выполняемые при попадании курсора мыши в определенную область экрана. Просто щелкните по интересующей вас области и выберите необходимое действие из выпадающего списка. Из верхнего левого угла хорошо работает Expo Edge. Он отодвинет текущий рабочий стол и покажет обзор всех ваших виртуальных столов. Щелкните правой кнопкой по любому из них, и вы переместитесь обратно. Для верхнего правого угла, возможно, неплохо подойдет Show Desktop. Как и в Apple'овском режиме Exposé, все открытые окна на текущем рабочем столе отодвинутся в сторону, и вы сможете добраться до расположенных на нем файлов и папок. Window Picker выполняет обратное действие, показывая миниатюры открытых программ, чтобы вы видели, в какую из них перейти.

Обновление дока

Взглянем правде в глаза: панель рабочего стола Gnome проигрывает тому пиршеству функций, что предлагает своим пользователям KDE. За последние годы его Gnome-эквивалент поменялся очень мало, и решение, как обычно и бывает в Linux, состоит в замене стандартной панели на нечто лучшее. Существует на удивление много жизнеспособных альтернатив, но *Cairo-Dock*, несомненно, лучшая замена панели Gnome из всех, что мы смогли отыскать. Существует масса причин, почему ему следует уделить внимание, однако это единственный док, имеющий рейтинг «умеренности» скачиваемых тем. И, как ни странно, ему так и надо! В большинстве дистрибутивов, за исключением Ubuntu, можно установить *Cairo-Dock* через менеджер пакетов. Пользователям Ubuntu для этого потребуется добавить вручную репозиторий 'repository.cairo-dock.org', обновить список пакетов, а затем установить *cairo-dock* и *cairo-dock-plug-ins*. Нацельтесь на версию 2.1, вышедшую 10 октября 2009 г. – более старые не дадут вам тех возможностей, о которых пойдет речь ниже.

При первом запуске программы она спросит, хотите ли вы задействовать OpenGL. Если на вашей машине *Compiz* работает без проблем, отвечайте «да», но если скажете «нет», то недосчитаетесь всего лишь нескольких эффектов и переходов. Если у вас установка о двух экранах, то вначале вы увидите ужасный док с рождественской темой, в центре обоих. Эту проблему легко решить, придержав Alt и перетащив мышью док в более подходящее место.

Чтобы сменить тему, просто щелкните правой кнопкой по доку и выберите меню *Cairo-Dock*, а затем Управление темами [Manage Themes]. Появившееся окно представит все темы, доступные на вашей системе, а на сайте www.cairo-dock.org можно скачать еще больше. Вам есть из чего выбирать – от идей, явно позаимствованных из других операционных систем, до дружественных к Gnome, и от красочных до унылых; ну, а нашей любимой темой стала 'Bret' от Benoit2600. Она использует почти монохромные иконки по дуге вокруг дока, и когда мышь оказывается над ними, они превращаются в маленькие вращающиеся кубики. Там даже есть *Cairo-Dock*-эквивалент для стеков из OS X: при удержании мыши над списком его элементы как бы выскакивают из исходной иконки. Добавьте к этому коллекцию четко прорисованных виджетов рабочего стола и глупую рыбку, которая плавает внизу экрана, и у вас будет выигрышная комбинация. LXF



Обратная связь

Мы будем рады узнать о ваших собственных улучшениях внешнего вида рабочего стола. Пришлите нам экранный снимок, а если мы накопим их достаточно много, то сделаем библиотеку самых красивых, чтобы дать людям представление о том, что можно сделать.



Блюститель

Корреспондент Linux Format встретился с Брэдли Куном, одним из разработчиков GPLv3. Брэдли считает, что плодить лишние лицензии ни к чему...

LINUX FORMAT Интервью

Бывший директор FSF, известный защитник GPL, автор пункта про Affero, Perl-программист и акула покера... мы знаем Брэдли Куна [Bradley Kuhn] в основном по его работе над лицензиями свободного ПО. Приняв участие в доводке текста GPLv3, Кун продолжает пресекать всяческие нарушения GPl. Не так давно он помог нескольким крупным проектам свободного ПО сформировать некоммерческие организации, миновав многочисленные бюрократические рогатки. Мы встретились с ним, чтобы обсудить процесс внедрения GPLv3, лицензионные исключения и важность взаимного соответствия лицензий...

Linux Format: Новейшая статистика Google показывает, что внедрение GPLv3 ускоряется, а вот у LGPLv3 наблюдается спад. Что вы думаете по этому поводу?

Брэдли Кун (БК): Меня это не очень беспокоит. Одни люди пользуются LGPL, другие – лицензией Apache. Причина в том, что «приватизировать» код Apache меньше желающих. Даже если лицензия позволяет такое сделать, не каждому этого



хочется. Я думаю, что с кодом Apache произошло вот что: люди понимают ценность коллективной работы на условиях этой лицензии, поэтому не хотят отгораживаться от мира. Для программистов LGPL имеет меньшую ценность: они не видят проблем, которые она призвана устранить. Есть много методов реализации принципа copyleft, и мы, те, кто работает в сфере «копилефта», не считаем, что наш путь единственно верный. Мы выбираем «копилефт» потому, что нам нравится юридическая защита свободного ПО. Но есть ведь и другие методы: сообщества BSD и Apache оказывают социальное давление по освобождению кода – и это работает! Я хорошо понимаю людей, избравших такую стратегию.

LXF: Нам кажется, некоторые проекты вообще никогда не перейдут на GPLv3 — например, ядро Linux.

БК: Вполне возможно. Но суждение по Linux еще не вынесено. Полагаю, сообщество и сам Линус [Торвальдс] все еще обдумывают этот вопрос. Да, некоторые проекты вряд ли переключатся. Но происходит любопытная вещь: ведь было время, когда GPLv2 считалась проблемной, и люди не хотели переходить на нее. Сам Линус говорил, что, если бы вернуть время вспять, он ни за что не стал бы трогать GPl. Работая над GPLv3, мы предчувствовали, что ее выпуск приведет к проявлению так называемого синдрома идеализации GPLv2. То есть те же самые люди, которым раньше не нравилась GPLv2, вдруг решат, что лучшей лицензии никогда и не было – а вот у GPLv3 сплошные проблемы. Я понимаю, что людям, не варившимся в гуще работы над GPLv3, нужно время на обдумывание. Не хочу, чтобы решения о смене лицензии принимались поспешно, без тщательной оценки последствий и подробного обсуждения в сообществе.

LXF: Вы считаете, что люди должны хорошенько поразмыслить над выбором лицензии — это понятно. Но согласитесь, что процесс выработки GPLv3 проходил весьма гладко.

БК: Не совсем. Многое приходилось увязывать с законодательством, а это всегда нелегко. Я недоволен тем, что пункт Affero пришлось выделить в отдельную лицензию. Бенджамин Мако Хилл [Benjamin Mako Hill] и я, вероятно, дольше всех пытались удержать поправку в GPL – как и было задумано с самого начала.

Начиная разработку GPLv3 в 2002 году, Эбен Моглен [Eben Moglen], Ричард Столлмен [Richard Stallman] и я держали совет. Основным вопросом был именно ввод пункта Affero. И мне кажется – а я занимался этим пунктом с первого дня – что он вошел бы в лицензию, так было задумано. Но из-за мощного противодействия многих людей, таких как Крис ДиБона [Chris DiBona] из Google, было решено составить отдельную лицензию и предоставить программистам самим выбирать, нужна ли она им. Элегантное решение. Я-то хотел сделать иначе, но и такой вариант меня устраивает: хакеры сами решили судьбу пункта Affero.

LXF: Добавление и удаление пунктов немного сбивает с толку, особенно если учесть, что в рамках GPLv3 программисты могут добавлять и удалять их сами.

БК: Типы поправок, которые можно добавлять – это исключения, а не добавочные ограничения. Структура исключений в GPLv3 полезна – они были всегда, их добавляли и в GPLv2, даже в кодовую базу GNU от FSF. Представление о том, что существует только четыре способа лицензирования – GPL, LGPL, Apache и BSD – не соответствует практике. Можно взять зазор между лицензиями (например, между GPL и LGPL) и несколько сократить его. Вы можете сказать: «Я хочу разрешить вещи, допустимые в рамках LGPL, но не все. Но мне не надо и "чистой" GPL: пусть будет нечто посредине». Применение исключений позволяет задать формат лицензирования с хирургической точностью.

› Кун доволен, что GPLv3 совместима с Apache 2.0.

LXF: Но, принявшись работать скальпелем, не угодим ли мы в итоге в стан лицензий Creative Commons?

БК: Не думаю, что это приведет к такому виду размножения лицензий, поскольку исключениями ведь можно и не пользоваться. Если вы смешиваете один продукт с исключениями, а другой – без них, исключения не действуют. Поэтому всегда есть общая база, к которой можно стремиться. При другом способе размножения лицензий возникает серия несовместимых документов, и можно зайти в тупик. В данном случае все лицензии сводятся к GPl. Я хотел бы подчеркнуть ее отличие от Creative Commons: это ортогональное пространство, поскольку они имеют дело не только с ПО. Они превосходно ориентируются в своем пространстве, и я рад, что они переняли принципы свободного ПО и распространяют их в других сферах.

Что такое пункт Affero?

GPL – и v2, и v3 – предписывает раскрывать изменения, сделанные в программе, только если вы ее распространяете. Проблема в том, что код, работающий через Интернет (например, популярная система управления контентом *Drupal*), никогда никому не передается. То есть любой человек может взять код, изменить, как ему захочется, и, не публикуя результаты, свободно пользоваться им. Пункт Affero – это модифицированная версия GPL, известная как AGPL; она требует, чтобы каждый, кто пользуется web-приложением по сети (например, в Интернете), имел доступ к его исходному коду.

LXF: Как вы считаете, хорошо ли, что GPLv3 теперь совместима с лицензией Apache 2.0?

БК: Теперь можно взять код Apache 2.0, скомбинировать его с кодом GPLv3 – и все будет нормально. Я думаю, это большой успех. Ирония заключается в том, что было время, когда многие предпочитали множить лицензии. Инициатива Open Source Initiative с момента зарождения старалась привлечь как можно больше людей под знамена открытого ПО и под открытые/свободные лицензии. Хорошо, что теперь они выступают против размножения лицензий. В значительной мере этот процесс заключается в расчистке сделанного ранее. Я возражал против размножения лицензий еще в 2001 году. Для пользы дела достаточно иметь минимальный комплект. LXF



Дифферент на нос!

Запуском Linux в воздух сегодня уже никого не удивишь, поэтому Антон Борисов решил идти за эксклюзивным материалом в прямо противоположном направлении.

Тема автономных подводных аппаратов (Autonomous Underwater Vehicle — AUV) приобретает все большую популярность. С одной стороны, всегда интересно посмотреть, что же происходит в «темных глубинах». С другой — комплекс полученных научных и статистических данных позволяет человеку приобрести немалый багаж знаний. Особенно приятно осознавать, что подобные аппараты сконструированы и работают под управлением Linux — и сегодня Эрин Фишелл [Erin Fischell] и Бенджамин Зейденберг [Benjamin Seidenberg] из проекта CUAUV объяснят, чем обусловлен этот выбор.

LXF: Эрин, давайте «начнем с начала»: что собой представляет команда CUAUV, и каковы ваши цели?

Эрин Фишелл (ЭФ): Группа по изучению дизайна и функционирования автономных подводных аппаратов университета Корнелл (CUAUV) – это студенческий проект, существующий уже лет десять. За это время командой было построено семь разных модификаций полностью работоспособных прототипов, участвовавших в разных соревнованиях и исследовательской работе. Подобные аппараты занимаются визуальным, акустическим и сонарным обнаружением предметов в водной среде. В частности, последние года три командой изготавливаются полностью различные модели – каждый год мы проектируем и спускаем под воду сильно отличающиеся друг от друга аппараты.

За десять лет у нас накопился приличный опыт в области построения надежных механических и электрических систем. Все ПО написано внутри университета, но используется и ряд открытых модулей. По мере усложнения разработок расширяется и круг задач, в которых используются спускаемые аппараты. Например, этим летом мы ездили в Итаку (штат Нью-Йорк) на озеро Каюга для изучения угроз загрязнения водной среды. И там с дизельного шлюпа Haendel мы провели ряд занятий. Сейчас же группа работает над новым аппаратом с кодовым названием «Akula» – будем надеяться, что получится привнести в этот прототип свежие мысли и наработки.

LXF: Корректно ли называть ваш проект любительской разработкой, чем-то вроде хобби?

Бенджамин Зейденберг (БЗ): Я считаю попытку выполнить такую работу силами энтузиастов нереальной. Прежде всего, требуются достаточно большие ресурсы. По большому счету, наш успех зависит от технической и материальной поддержки наших спонсоров и, конечно же, от неоценимой поддержки родного университета. Например, что касается времени: 35 членов группы затрачивают на проект в среднем от 15 до 30 часов в неделю каждый. Мне кажется, что в качестве хобби такой режим работы невозможен. Впрочем, если кто-то надумает этим заняться, то некоторые полезные сведения можно найти здесь: http://cuauv.org/node/1015.

› **CAD-модель создается в самом начале нового проекта.**

Что внутри?

Первые модели CUAUV конструировались на базе материнских плат VersaLogic (128 МБ ОЗУ и 400-МГц AMD K6-2). Использовалось Linux-ядро версии 2.3 с патчами реального времени. Дальнейшие модели комплектовались уже двумя процессорами – Transmeta Crusoe отвечал за основное движение и навигацию, Intel Pentium III – за обработку видео и распознавание образов. В современных моделях (Triton и Nova) – более мощные вычислительные блоки на базе Intel Core 2 Duo, их хватает на все потребности подводного аппарата.

LXF: А вы когда-нибудь оценивали примерную стоимость разработки?

ЭФ: Да, каждый новый год мы проводим анализ стоимости материалов, которые придется купить. Если бы не спонсорская помощь, то одни сенсоры и двигатели потянули бы на 75000 долларов, а наш реальный, рабочий бюджет – всего около 20000. Для проведения разъездных мероприятий у нас отдельная статья расходов, не включенных в бюджет. В бюджет также не входят трудовые затраты – работа наших студентов происходит на добровольных началах. Помимо прочего, университет бесплатно предоставляет нам как место для занятий, так и инструментарий.

LXF: Некоторое время назад вы заняли первое место в конкурсе International Autonomous Underwater Vehicle Competition в Сан-Диего (LXF124). По-вашему, каким требованиям должен удовлетворять удачный проект в этой области?

ЭФ: Самым главным фактором успеха явилось очень тщательное тестирование. Все модули электрических и механических подсистем много раз прогонялись на стендах, прежде чем были вмонтированы в аппарат. Безотказная работа указанных подсистем позволила потратить все лето на проверку работы ПО. В среднем, получилось где-то 150 часов ходовых испытаний под водой. В целом, секрет успеха кроется в хорошем дизайне и тестировании.

LXF: Почему вы выбрали Debian, а не, скажем, Red Hat или SUSE, или даже FreeBSD?

БЗ: Debian мы выбрали за его гибкость и простоту использования. И, главное, за стабильность. Несколько лет назад я был одним из тех, кто принял это решение, так как я отвечал за программную платформу. Сказался также опыт работы с Debian и многочисленные связи в Debian-сообществе.

LXF: Планируете ли вы выложить «исходный код субмарины», чтобы сообщество вносило дальнейшие улучшения?

БЗ: Большая часть кода очень специализирована и специфична именно для нашей платформы. Например, у нас есть специализированный блок аппаратуры, с которым работает наше ПО. Такого рода код не будет интересен никому, кроме нас. К тому же, наша главная цель – выиграть соревнования. И мне кажется, что передача кода в открытый доступ нас к ней не приблизит.

Справедливости ради отмечу, что когда мы находим в свободном коде ошибки или недочеты иного рода, то присылаем свои изменения разработчикам. Например, я контактировал с разработчиками OpenCV-пакета из Debian, когда потребовалось исправить недочеты, связанные с захватом изображения с камеры.

LXF: А каким образом сообщество может быть полезно в ваших будущих разработках?

БЗ: Естественно, разрабатывай мы все компоненты с нуля, на создание субмарины ушло бы куда больше времени. Использование открытого инструментария, например, *GEDA* (для проектирования электроники), *OpenCV* (машинное зрение) и еще многих дополнительных библиотек и утилит позволило нам значительно сократить время на разработку. Так что в этом плане мы (да и другие энтузиасты-разработчики) многим обязаны свободному сообществу.

LXF: А на свободное ПО завязан весь производственный цикл, от стадии CAD-проектирования до программирования?

БЗ: К сожалению, нет. Вся электрика и программная часть спроектирована на свободном ПО, но вот для целей механики используется коммерческая моделирующая CAD-программа.

LXF: Какие практические цели ставятся перед прототипом Nova? И на какие глубины он может опускаться?

ЭФ: CUAUV в данный момент сотрудничает с проектом Cayuga Lake Floating Classroom Project (Итака, Нью-Йорк) в рамках изучения экологии в озере. За лето аппаратом было сделано три погружения на юго-западе озера и сфотографирован процесс роста водорослей-макрофитов. Данные фотографии позволяют оценить хронологию процесса вегетации за лето. Помимо непосредственно процесса видеосъемки, Nova записывает данные с температурных датчиков, так что на руках у нас есть целая картина по течениям и их направлениям в разное время. Хотя на аппарате Nova установлены только видео- и температурные датчики, у нашей команды есть возможность добавить и ряд других, с тем, чтобы была возможность исследовать большие глубины. В частности, был специально спроектирован отсек для глубоководной камеры с возможностью подключения оптического кабеля и наблюдения в реальном времени. В какой-то степени, Nova – это первая платформа, спроектированная CUAUV для работы в открытой воде (море), со множеством исследовательских возможностей.



› Программный дизайн аппарата.

LXF: Максимальная глубина погружения, я полагаю, 30–50 метров?

ЭФ: Тесты в озере проводились на глубинах от 5 до 15 метров, где, собственно, и происходит рост растений. Мы доходили до отметки 30 метров, а вот в следующем году будем проводить испытания уже на 30–50 метрах.

LXF: Возвращаясь к применениям аппаратов в повседневной жизни: были ли у вас контакты с организациями, которые хотели бы использовать ваши разработки в бизнес-задачах?

БЗ: В прошлом мы работали с некоторыми компаниями, тестируя или создавая различные узлы для их нужд. Например, нам дали микродвигатели со значительной скидкой; в обмен мы поделились разработанной нами платой управления для них и положительно отозвались о данной продукции. На нас также выходят организации, которые хотели бы выполнять совместные исследовательские проекты.

В целом, наши субмарины – это специализированные платформы для отработки AUV-технологий. И, по большому счету, они не слишком приспособлены для чисто коммерческого применения. Особенно на фоне компаний, которые продают бизнес-аппараты. Мы все-таки исследователи. LXF

› Каждый год для соревнований проектируется новая модель.



Все идет по Плану

Каким получился бы Unix, если бы его придумали лет на двадцать позже? Ответ на этот вопрос попытается найти **Роман Ярыженко**.

LINUX FORMAT на DVD
» Plan 9 (ISO-образ)



Наш эксперт

Роман Ярыженко
Всегда готов потратить время и силы на изучение новой операционной системы – особенно если она распространяется вместе с исходными текстами.

Plan 9 из открытого косм... Ой, простите! Plan 9 из Bell Labs. Да-да, той самой Bell Laboratories, из которой вышли языки C/C++, Unix и многое другое, без чего современные компьютерные науки были бы не совсем науками или совсем не науками. Но, оказывается, из стен Bell Labs вышел не только Unix. Были разработаны еще две ОС, одна из которых, Plan 9, послужила причиной для написания данной статьи. На территории России, да и у себя на родине, в Соединенных Штатах, Plan 9 не пользуется широкой известностью, и мы решили попытаться исправить этот недостаток хотя бы среди читателей LXF.

Немного истории

В 80-х годах сотрудникам Bell Labs, а именно Брайану Кернигану [Brian Kernigan], Робу Пайку [Rob Pike] и некоторым другим, показалось, что система Unix... недостаточно подходит для рабочих станций! Доводы были весомыми. Во-первых, Unix – старая система, разработанная для мэйнфреймов и плохо приспособленная для концепций, появившихся после них (таких, например, как сети или графическая система). Во-вторых, Unix с некоторых пор стал закрытым, и это вызвало некоторые затруднения (например, стало сложнее писать драйверы) Ну, а главное – у компьютеров появилась приставка «персональные», и, соответственно, сетями стало нелегко управлять централизованно.

> «В Plan 9 первыми сумели реализовать поддержку UTF-8.»

В середине 80-х в Bell Laboratories началась работа над проектом, устранявшим вышеперечисленные недостатки, которым был подвержен Unix. В 1989 году система стабилизировалась настолько, что стала использоваться в Bell Labs в качестве единственной. Первая публичная версия (доступная для вузов США) вышла в 1992 и получила название Plan 9, в честь некогда популярного в Америке фильма «Plan 9 from Outer Space» («План 9 из открытого космоса»).

Изначально Plan 9 поддерживал платформы Sun SPARC, SGI Power и Magnum с NeXT в качестве CPU-серверов (см. ниже), а также PC в виде терминалов. Первая версия включала такие вещи, как текстовый редактор *sam*, язык Alef и оконная система *8½*, и самое главное – в Plan 9 первыми сумели реализовать поддержку UTF-8; при этом, в отличие от многих современных ОС, Unicode в нем был основным набором символов, и вся внутренняя обработка осуществлялась на нем!

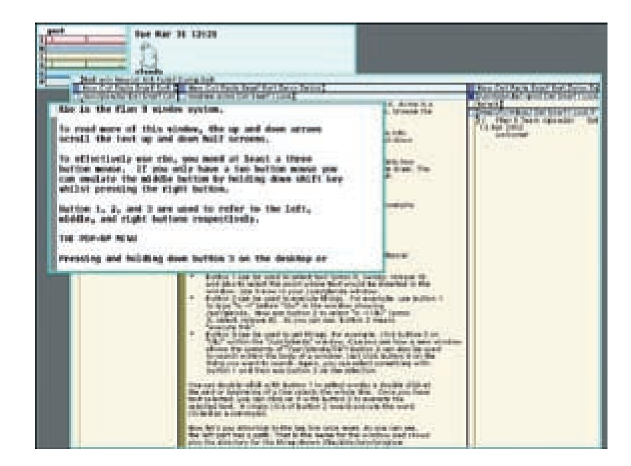
Вторая версия Plan 9 вышла под именем Brazil в 1995 году; в 1999 исходное название вернулось. Она была доступна в двух вариантах: коробочном с CD ($350), содержащем, кроме дистрибутива, исходные коды и печатные руководства, и бесплатном, на четырех дискетах, но без исходников. В Plan 9 добавили среду разработки *acme* и несколько утилит. *8½* был переписан на Alef.

Третья версия, бесплатная для некоммерческого применения и распространявшаяся вместе с исходными текстами, появилась в 2000 году. В ней произошел отказ от Alef, так как расходы на поддержку сразу нескольких языков оказались обременительными. Появились порты на несколько новых платформ. Систему *8½*, переписав с Alef на C, немного усовершенствовали и переименовали в *Rio*. Появилась поддержка устройств PCI и PCMCIA.

Четвертая, последняя полная версия – она была выпущена Bell Labs/Lucent – появилась в 2002 году. Она распространялась по свободной лицензии Lucent Public License 1.02, одобренной OSI и FSF. Изменения претерпел файловый сервер: в частности, он стал понимать длинные имена. Обновился и агент безопасности – *factotum*.

После четвертой версии разработчики перешли на инкрементальное версионирование, но далее в статье будет рассматриваться «каноническая» четвертая версия; к тому же большинство сборок для конечного пользователя являются абсолютно одинаковыми.

Но, по-видимому, читателю уже наскучило нудное перечисление дат и особенностей релизов? Тогда – вперед, будем ставить Plan 9!



› **Так выглядит *Rio*, рабочий стол Plan 9, сразу после загрузки.**

Поехали!

Перед установкой хотелось бы дать несколько советов. Во-первых, Plan 9 может испытывать проблемы с запуском в ряде современных виртуальных машин (*VMware*, *VirtualBox*). Мы для знакомства с ним воспользовались старым компьютером, но если у вас под рукой такого нет, возьмите *QEMU*. Во-вторых (как бы странно это ни звучало), Plan 9 не является еще одним вариантом Unix. Да, некоторые вещи похожи, но пусть это не вводит вас в заблуждение. А посему – забудьте на время многое из того, что вы знаете о Unix/Linux. Ну, и в-третьих – ничего не бойтесь. Ах, да! Еще очень рекомендуется трехкнопочная мышь.

Непосредственно процесс установки иллюстрирован во врезке справа. Внимательно пройдите все шаги, следуя инструкциям инсталлятора. Вопросы здесь может вызвать только деление на разделы и подразделы, напоминающее слайсы во FreeBSD. Для этого используются программы *disk/fdisk* и *disk/prep*, вызываемые

в процессе установки автоматически. В обеих программах выход с сохранением – **w**, затем **q**, как в *Vi*.

Давайте познакомимся?

Загрузчик в Plan 9 чем-то похож на старый добрый *LILO*, за одним приятным исключением – его не надо обновлять после изменения конфигурационного файла. Он, как и ядро, размещается на отдельном разделе, который называется 9fat и, по сути, является обычным разделом FAT16. Кстати, ядро в Plan 9 монолитное и реализует несколько десятков системных вызовов.

После старта ядра нужно ответить на вопрос, откуда загружаться дальше – по сети или с локального компьютера. По умолчанию выбрана малопонятная запись: «local!#S/sdC0/fossil». Расшифруем: local – локальный компьютер, #S – внутреннее название каталога с жесткими дисками (используется, когда еще не сформировано единое пространство имен), sdC0 – обозначение диска (sdNY, где N=C, D, ... – первичный, вторичный, ... , ATA-контроллеры, Y=0,1 – ведущий/master или ведомый/slave), fossil – имя раздела. Непосредственно диски в Plan 9 являются каталогами, файлы же в этих каталогах – за исключением нескольких – это разделы. К «сырым» данным всего диска тоже можно обратиться, для чего в каталогах есть файл по имени **data**.

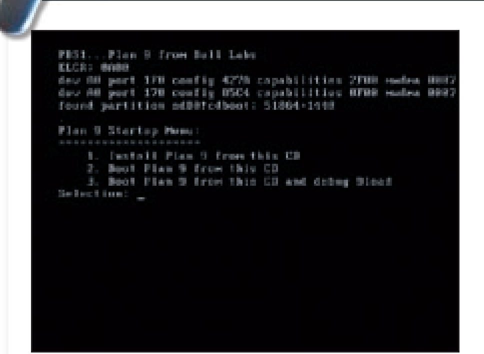
Нажав Enter, вы увидите запрос имени пользователя. Не пытайтесь ответить 'root': в Linux это не рекомендуется по соображениям безопасности, а в Plan 9 такого просто нет! Теперь вас зовут glenda (в честь талисмана Plan 9). Кстати, Plan 9 на автономном компьютере фактически является однопользовательской ОС. Чтобы использовать все преимущества многопользовательского Plan 9, нужны как минимум два соединенных компьютера; на одном из них будет запущен сервер аутентификации (его часто объединяют с файловым сервером, но последнее не обязательно).

Итак, вы вошли в систему. Спустя пару секунд запустится *Rio* (см. рисунок на предыдущей странице). Вы увидите несколько окон. На переднем плане – терминал. Вверху справа – *stats*, местный системный монитор. На заднем плане – *acme*, аналог *Emacs*.

В Plan 9 используются все три кнопки мыши. Первая нужна для выделения текста, вторая – для выполнения операций над ним (вырезать, вставить, ...), третья – для управления окном (создать, скрыть, удалить, ...). Если нет второй, средней, кнопки, ее можно имитировать нажатием Shift+третья. Инициация и подтверждение действий выполняются одной и той же кнопкой: например, чтобы закрыть окно, надо щелкнуть правой кнопкой по свободному (серому) фону, выбрать Delete, отпустить кнопку, навести курсор на окно и щелкнуть той же самой правой (не левой!) кнопкой мыши.

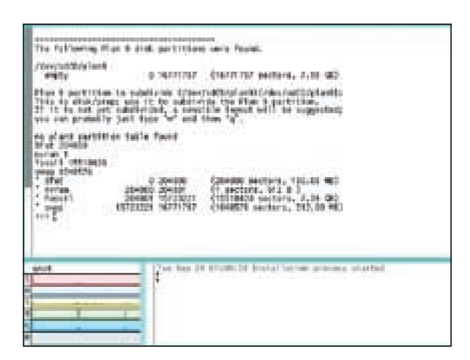
В Plan 9 все есть файл. Нас с вами этим не удивишь, но здесь знакомая концепция доведена до предела: сокеты, ioctl(3), fcntl(3) и прочие «сложности» банально отсутствуют. За все надо платить, и ценой за простоту программирования в Plan 9 является несовместимость со стандартом POSIX. Есть, конечно, так называемая «среда ANSI/POSIX» (ANSI/POSIX Environment, APE), есть даже порты некоторых Unix-приложений (*Vi*, *X11*) под нее, но сомнительно, что здесь удастся скомпилировать сколько-нибудь современную программу. »

Шаг за шагом: Устанавливаем Plan 9



1 Загрузка

Этот экран вы увидите сразу после загрузки. Выберите первый пункт, если хотите установить Plan 9. Остальные опции предназначены для старта в режиме LiveCD.



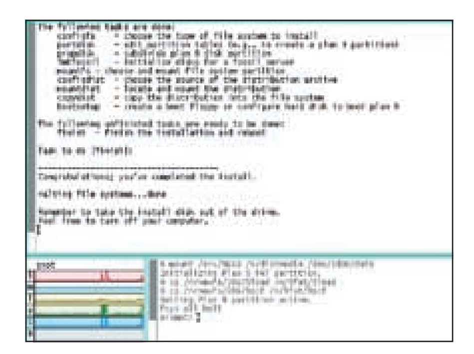
2 Разбиение диска

Для установки Plan 9 необходимо создать первичный раздел командой **t pN plan9**, где N – номер раздела от 1 до 4, plan9 – тип раздела. Внутри него создаются подразделы.



3 Копирование файлов

Пока происходит копирование файлов дистрибутива, можно по традиции откинуться на спинку кресла и выпить чашечку кофе.



4 Готово!

Все! Установка завершена. Теперь можно извлечь диск из виртуального или реального привода и перезагрузить компьютер.

В полной мере возможности Plan 9 раскрываются в распределенной сети. Рассмотрим вкратце ее иерархию:

» CPU-сервер – используется для запуска неинтерактивных программ и приложений, требующих вычислительной мощности.

» Сервер аутентификации – предназначен для регистрации с терминалов пользователей, обычно объединен с файловым сервером.

» Файловый сервер – машина для хранения файлов пользователей. В Bell Labs в качестве устройств хранения использовались WORM-накопители.

» Терминалы – пользовательские компьютеры.

Кроме того, в качестве необходимого «довеска» к вышеперечисленному, в Plan 9 есть пространства имен, придающие несколько иной смысл стандартной утилите *mount*, и команда *bind*.

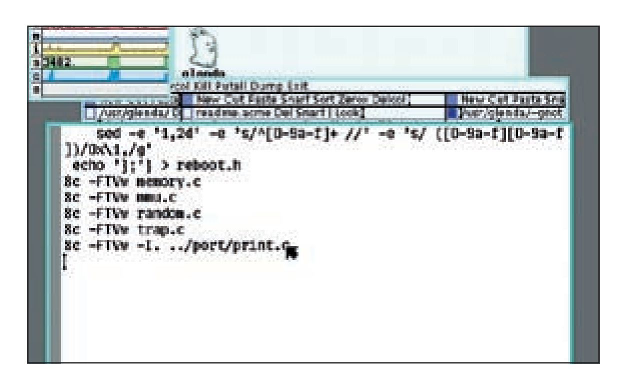
Пространством имен в Plan 9 называется вся древовидная структура файлов (не забывайте, что файлом в этой ОС может быть что угодно), видимая пользователем. Таким образом, оно может охватывать несколько компьютеров.

Команда *mount* подключает к каталогу файловый сервис. У нее есть параметр, определяющий способ подключения: либо заменить текущее содержимое каталога новым, либо появиться перед текущим содержимым, либо после него. Команда *bind* объединяет два каталога в один. Параметры (и даже man-страница) с командой *mount* у нее общие.

Наше мнение

Plan 9 – так вполне мог бы выглядеть Unix, будь он создан немного позже. Но история не терпит сослагательного наклонения, и Unix в таком виде, в каком он появился, влияет на существующее положение вещей и поныне. Конечно, с тех пор в нем много чего изменилось и добавилось, однако ничего революционного за протекшие лет 20 в его развитии не произошло (трехмерные рабочие столы не в счет: мы говорим о фундаментальных вещах). Исходя из вышесказанного, можно подумать, что вся компьютерная эволюция застряла в 80-х. Тем не менее, это не так. Plan 9 содержит много нововведений, которые даже сейчас, спустя более чем 20 лет с момента его появления, кажутся удивительными. И это при том, что многие из них очень просты и изящны.

Plan 9 – одна из немногих операционных систем, не только вышедших за стены лабораторий, но и используемых в коммерческих целях (скажем, в той же Alcatel-Lucent). Это доказывает, что он весьма практичен и зрел.



› В Plan 9 тоже можно собрать свое ядро. Компилятор называется *8c* — такова схема именования, перенятая Пайком и для Google Go (go-lang.org).

Слово Vita Nuova

Мы посчитали, что знакомство с Plan 9 будет неполным без обмена мнениями с «живым разработчиком».

И поэтому отыскали в Сети д-ра Чарльза Форсита [Dr Charles Forsyth], одного из основателей и технического директора компании Vita Nuova (www.vitanuova.com), а также убежденного сторонника Plan 9, использующего его с начала 90-х годов. Он любезно согласился ответить на все наши вопросы.

LXF: Доктор Форсит, чем занимается ваша компания и как она связана с Plan 9?

Д-р Чарльз Форсит (ЧФ): Vita Nuova была основана в середине 1996-го года для работ в области сетей и распределенных вычислений; при этом использовался «младший брат» Plan 9 – Inferno. Конечно, и Plan 9, и Inferno тогда были платными; тем не менее, они мне нравились, и я перешел из Йоркского университета, где работал в то время, в Vita Nuova.

› Д-р Чарльз Форсит — справа.

LXF: А Vita Nuova — единственный лицензиар Plan 9?

ЧФ: Нет. Plan 9 из Bell Labs был выпущен под лицензией Lucent Public License, и его лицензиарами были сначала AT&T, затем Lucent, а сегодня его лицензирует Alcatel-Lucent. Мы распространяем печатную документацию для четвертой версии Plan 9. Данная версия является последней полной: после был переход на инкрементальное версионирование, как это сделано во многих других свободных/открытых проектах. Раньше мы распространяли и CD, но потом нужда в этом отпала; сейчас пользователям проще скачать продукт с сайта.

Вначале в Vita Nuova использовали Plan 9 для внутренних нужд, потом, доработав, мы начали его продавать.

LXF: А какую, по-вашему, цель преследовало создание Plan 9?

ЧФ: ОС Plan 9 готовилась Bell Labs в качестве новой платформы, объединяющей в себе как программную, так и аппаратную части. Система разрабатывалась «с нуля» с упором на распределенность и параллелизм.

В Plan 9 применены как старые, так и новые идеи в сфере распределенных вычислений. Это отличает его от его предшественника, Unix (включая и его потомков, BSD и Linux), модель которого представляет собой одиночную замкнутую машину с локальным хранилищем данных.

LXF: Наш журнал пропагандирует открытое и свободное ПО, и потому мы не можем не спросить: что вы думаете по этому поводу? Как в связи с Plan 9, так и в целом.

ЧФ: В моем понимании, свободное ПО было придумано еще до FSF. Во времена моей молодости было свободное распространение полезных программ от DECUS (Digital Equipment Computer Users Society) и от подобных организаций для других платформ. Это было похоже на обмен ПО между университетами: дружественная атмосфера, обширное распространение ПО в виде исходных кодов, без искусственных ограничений – как для использования, так и для экспериментов.

Современное изобилие свободного ПО, конечно же, затрудняет зарабатывание денег на продаже программных пакетов. Даже «нишевые» приложения теснятся под напором свободного ПО. Альтернатива – предоставление поддержки продуктов. Vita Nuova зарабатывает деньги на технических сервисах, а не на продаже ПО или документации. Мы, кроме того, заботимся не только о своем, но и о чужом открытом ПО, отчасти потому, что успели адаптироваться раньше других.

Свободное ПО кажется сейчас хорошим выбором для рабочей среды. С другой стороны, несмотря на то, что некоторые люди из СПО считают себя радикальными, многое из используемого мной в Ubuntu не является революционно новым с точки зрения архитектуры (некоторые дополнительные пакеты все же отражают больше инноваций). Одной из захватывающих вещей в Unix 70-х была его простота по сравнению со сложными и раздутыми операционными системами того времени. Сейчас же, переключаясь с Plan 9 на программирование под Linux/BSD, я дивлюсь, как они консервативны и избыточно сложны.

LXF: Если не секрет, кто (и где) использует Plan 9?

ЧФ: В списке рассылки 9fans есть много энтузиастов, которые экспериментируют с Plan 9, и некоторые из них используют его в качестве основной среды разработки, в том числе и в коммерческих организациях. Не секрет, что Vita Nuova многие годы тоже применяет его как платформу для разработки ПО. Чего стоит одна файловая система dump! Она автоматически сохраняет состояние целой системы, дополнительно записывая изменения в службу архивов Venti. Она предшествовала Apple Time Machine, но является более качественной – об этом мне говорили люди, которые используют обе ФС.

Более интересны группы, использующие Plan 9 в коммерческих приложениях. Вы правы насчет того, что некоторые коммерческие пользователи не говорят, что именно они применяют (мы знаем о них только потому, что они заключают с нами договор о предоставлении услуг). Но я все же могу привести пару примеров нетривиального коммерческого использования нашей системы.

» Компания Coraid (www.coraid.com) эффективно применяет урезанную и доработанную ею версию Plan 9 в популярных сетевых хранилищах данных.

» Alcatel-Lucent сама использует Plan 9 (и его возможности операционной системы реального времени) в коммуникационной сфере, например, в беспроводных базовых станциях.

LXF: Вы упомянули операционные системы реального времени…

ЧФ: Да, все верно: в Plan 9 есть планировщик реального времени и еще некоторые другие специальные механизмы RTOS, такие как неблокирующие семафоры и очереди, не требующие ожидания (используя их, процессы реального времени могут предотвращать бессмысленные блокировки в обычных процессах). Под «процессами реального времени» я имею в виду процессы, имеющие жестко заданный срок окончания работы, а не только быстрые или высокоприоритетные.

Примитивы реального времени присутствуют в Plan 9 для поддержки некоторых специализированных коммуникационных приложений, но с успехом могут использоваться, например, для мультимедиа-проигрывателя. Если вас всерьез интересует данный вопрос, советую обратиться к статье Сэйджа Маллендера [Sage Mullender] и Джима Мак-Ки [Jim McKie], касающейся большинства перечисленных выше вопросов (http://lsub.org/iwp9/cready/realtime.pdf).

LXF: Возможно ли использование Plan 9 во встраиваемых системах?

ЧФ: Да. Более того, это, вероятно, одно из самых частых его применений. Он удобен и как компьютер разработчика, но, к примеру, не слишком полезен в традиционной офисной среде.

LXF: Можно ли собрать ядро Plan 9 таким образом, чтобы появилась возможность подключать/отключать драйверы «на лету»?

ЧФ: Многие вещи, которые в большинстве других операционных системах реализованы в ядре, в Plan 9 являются приложениями режима пользователя. Многие сервисы в Plan 9 предоставляются файл-серверами, которые являются обыкновенными пользовательскими программами, реализующими протокол 9P.

LXF: Мы несколько раз упомянули, что в основе Plan 9 заложены оригинальные архитектурные решения. А какие именно?

ЧФ: Plan 9 является достаточно стандартной системой с примерно 38 системными вызовами [согласно заголовочным файлам – чуть более 50, – *прим.* **LXF**], что может показаться излишним, но некоторые из них (например, управление сегментацией памяти) довольно специализированы и применяются не очень широко. Основой параллелизма является процесс, и процессы, как с разделяемой памятью, так и без нее, создаются с помощью примитива rfork, который позволяет управлять также разделением других ресурсов, таких как файловые дескрипторы и пространства имен. Еще один примитив, rendezvous, используется для синхронизации. Все остальное – спин-блокировки, блокирующие очереди и каналы – реализованы в библиотеках режима пользователя через rendezvous. Большая часть остальных системных вызовов – операции над файловыми дескрипторами (read, write, seek и close) и операции над иерархическим пространством имен (open, create, chdir, stat, wstat, remove, bind, mount и unmount). В Plan 9 есть только один тип процесса (то есть система не делает различия между легковесными «потоками», которые разделяют память, и тяжеловесными «процессами», которые этого не делают).

Plan 9 необычен тем, что системные сервисы представляются и доступны как файлы в иерархическом пространстве имен. Все сосредоточено с помощью единого протокола разделения файлов, 9P.

О КЛАССИЧЕСКОМ UNIX

«Переходя с Plan 9 на Linux/BSD, дивлюсь, как они сложны.»

LXF: Ну и напоследок позвольте спросить, какие у Vita Nuova планы относительно будущего Plan 9?

ЧФ: Сейчас мы очень тесно связаны со многими другими компаниями (скажем, Bell Labs, IBM Research и Sandia National Labs) в проекте для использования Plan 9 в суперкомпьютере IBM Blue Gene, который включает адаптирование системы для поддержки приложений на так называемых «системах петамасштаба». Мы ожидаем, что это скажется и на применении Plan 9 в «обычном мире» – в частности, для работы в сети и на файловых хранилищах. LXF

Пять советов



Наш эксперт

Александр Толстой Коллекционирует полезные советы, которые помогают ему сделать свой Linux не таким, как у других.

Каждый линуксоид со временем накапливает собственную подборку хитростей. **Александр Толстой** готов поделиться с вами пятью лучшими из своей коллекции.

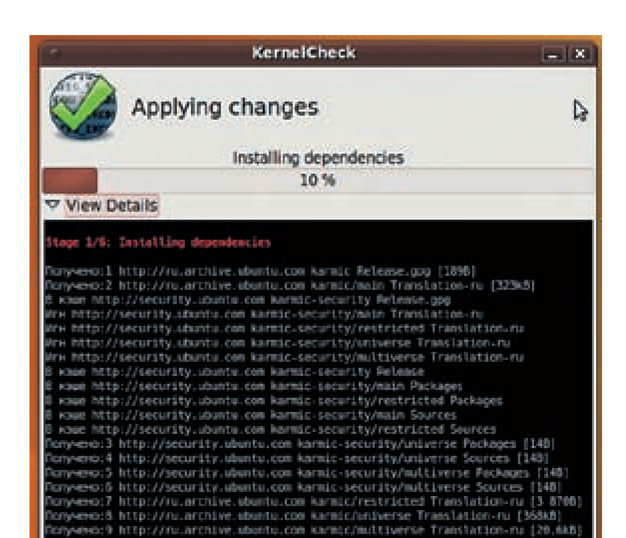
Как мы (и не только мы) уже неоднократно отмечали, быть просто пользователем Linux можно, но неинтересно. Статья «Ускоряем Linux» из ноябрьского номера журнала (LXF124) вызвала оживленную дискуссию как на форуме, так и в редакционной переписке, и мы решили, что тема заслуживает продолжения. Однако повышение производительности системы — одна из самых важных, но далеко не единственная задача тюнинга, поэтому сегодня мы также сконцентрируемся на трюках, которые помогут вам выполнять известные задачи с большим удобством или просто по-другому.

Ядро без команд

Несмотря на уверения, что «это несложно» и «под силу даже новичку», пересборка ядра по-прежнему подвластна лишь тем, кто сумел одолеть «кандидатский минимум» в Linux. Не то чтобы это было плохо, или иметь собственное ядро было бы насущной необходимостью – к слову, мы в Башнях **LXF** не испытываем таковой вот уже несколько лет – но возможность сделать эту процедуру чуточку проще никогда не помешает.

KernelCheck (http://kcheck.sf.net) – это графический интерфейс к набору сценариев на Python, берущих на себя всю «грязную» работу по перекомпиляции ядра для дистрибутивов на базе Debian. Всё верно: вы можете собрать и детально настроить свое собственное ядро Linux, не введя ни одной команды в терминале. *KernelCheck* предоставляет в ваше распоряжение пошаговый мастер, который готов провести пользователя по всем этапам выбора, загрузки и сборки ядра.

> «KernelCheck позволит делать чёрную работу в белых перчатках.»



› *KernelCheck* предлагает для сборки ядра интерфейс в виде мастера.

На первом экране следует нажать на кнопку скачивания сведений о свежей версии ядра. Данные берутся с сайта http://master.kernel.org и позволяют вам сравнить версии текущего стабильного и нестабильного ядер, узнать об актуальных наборах патчей, которые вы опять же можете применить или проигнорировать (в последнем случае получится так называемое «ванильное» ядро – по аналогии с ванильным мороженым, к которому можно добавлять различные наполнители). Сделав и подтвердив свой выбор, вы переходите далее, и *KernelCheck* начинает готовить для вас среду сборки: скачивает и устанавливает все необходимые пакеты разработчика. На свежеустановленной системе Ubuntu это напоминает установку пакетов из группы *build-essentials*, но только чуть удобнее и прозрачнее для пользователя.

Когда всё будет готово, *KernelCheck* запустит для вас интерфейс *xconfig* для настройки параметров ядра. Конфигурация по умолчанию берется из «фабричных» настроек вашего дистрибутива и копируется в файл **/usr/src/linux/.config**. Все сделанные вами изменения будут записаны в него же.

Сохранившись и выйдя из *xconfig*, вы сразу же будете перехвачены поджидающим вас скриптом из *KernelCheck*, который отвечает за сборку самого ядра, подгружаемых модулей и первичного образа загрузки (initrd). Теперь нужно подождать: процесс сборки ядра долог, и даже на быстрых машинах может занять около часа. Отдохнув от компьютера и дав поработать *make*, вы через какое-то время получите готовое ядро. *KernelCheck* упакует весь результат в два deb-пакета (само ядро и заголовочные файлы) и установит его обычной командой *dpkg*. При последующей загрузке новое ядро появится в списке *Grub*, и вы сможете запустить систему «с него». *KernelCheck* удобен и в том, что в дальнейшем вы сможете работать с установленными у вас исходными кодами ядра, обновляясь до будущих версий с помощью наборов патчей (загружать всё по новой не требуется), а также управлять версиями ядра без копания в конфигурационных файлах. *KernelCheck* позволяет делать чёрную работу в белых перчатках – за это его и любят продвинутые пользователи Debian-систем. Осталось дождаться версии программы для RPM!

Speed Disk для Linux

Дефрагментация в Linux? Зачем?! Почти все знают, что применяемые в Linux файловые системы, включая ext3, ext4, ReiserFS и XFS, невероятно устойчивы к фрагментации записанных данных. Так и есть, но всё же бывают и частные случаи, когда дефрагментация пригодилась бы. Скажем, у вас есть выделенный раздел для каталога **/boot**, и вы регулярно пересобираете ядра. Или вы создали специальный раздел для хранения часто обновляемых данных. Прошло два года – и вы замечаете, что чтение и запись на раздел сильно замедлились. Что делать?

Во-первых, размонтируйте раздел и запустите проверку диска командой *fsck* (если речь идет о корневом разделе, удобно воспользоваться LiveCD). Чтобы *fsck* проделала полную проверку даже исправного раздела, используйте параметр **-f**, например, так: **fsck -f /dev/sda1** (вместо sda1, разумеется, подставьте требуемое имя раздела). Нас интересует доля фрагментированных файлов, которые в данном случае называются 'non-contiguous'. Если фрагментация не превышает 15%, то можно не волноваться; если же она больше – нужна дефрагментация. Если говорить в целом, то для Linux нет специализированных дефрагментаторов, поэтому на большинстве форумов вам посоветуют сохранить все данные на другой раздел или носитель, отформатировать исходный раздел, после чего записать все данные обратно. Это долго и неудобно; но, к счастью, есть альтернатива. Некто Кон Коливас [Con Kolivas], некогда автор ветки -ck ядра Linux, создатель нового планировщика и прочая, и прочая, написал замечательный скрипт, который распределяет файлы на разделе от большого к малому, то есть сортирует их. Сценарий доступен для свободной загрузки по адресу: http://ck.kolivas.org/apps/defrag/. Особенность его в том, что он действует на всё, что находится в текущей директории, то есть позволяет дефрагментировать отдельные каталоги.

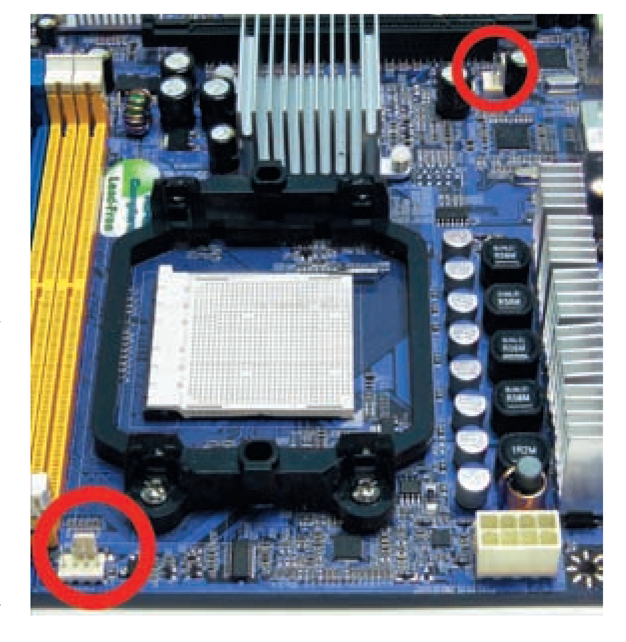
Если у вас раздел с файловой системой XFS, то дела обстоят по-другому. Вам потребуется программа *xfsdump* (если её нет в репозитории вашего дистрибутива, то исходный код можно скачать с сайта http://ftp.de.debian.org/debian/pool/main/x/xfsdump/xfsdump_3.0.2.tar.gz). Выполните от лица администратора команду **xfs_db -r /dev/sda1**, причем раздел sda1 предварительно размонтировать не нужно. Вы увидите значение параметра 'Fragmentation factor'. Для дефрагментации используйте команду: **xfs_fsr -v /dev/sda1**. Вот и всё! Кстати, если выполнить команду *xfs_fsr* без параметров, то она автоматически начнёт дефрагментацию всех смонтированных XFS-разделов. Если вы нетерпеливы и прервёте работу дефрагментатора, то при следующем запуске *xfs_fsr* возобновит работу, учитывая уже проделанные действия.

› На скорость дефрагментации влияет в основном не количество файлов, а их размер: на всю файловую систему может и полдня уйти.

Кино в тишине

Настоящий линуксоид – тоже человек, и иногда он хочет расслабиться и посмотреть какой-нибудь хороший фильм. Чтобы просмотр кино был комфортным, нужен не только большой экран и колонки – требуется ещё, чтобы системный блок не издавал шума, что приблизило бы его к бытовой Hi-Fi-технике и сделало бы более эргономичным. Разберёмся: в системном блоке шумят вентилятор на процессоре, винчестер, вентилятор блока питания и, в некоторых ситуациях, вентиляторы видеокарты и чипсета материнской платы. Уровень шума в каждом случае будет разный: где-то на ЦП установлен мощный вентилятор, который крутится медленно и мало шумит, у кого-то нет активного охлаждения чипсета и видеокарты. Параллельно с этим не будем забывать, что современные материнские платы автоматически регулируют скорость вращения системных вентиляторов, однако на шум это влияет мало – производители перестраховываются и не позволяют своей технике быть слишком тихой. В таких случаях как раз пригодится ручная настройка. Мы начнем с вентилятора процессора, для чего воспользуемся утилитами из состава *lm-sensors*.

Команда *sensors* позволяет выяснить, какие датчики имеются на вашей материнской плате и какой из них отвечает за управление обдувом ЦП. После этого запустим *pwmconfig*, который поможет



› На современных материнских платах таких разъемов не менее трех. Отрегулировав напряжение на них, можно существенно снизить уровень шума.

откалибровать напряжение, подаваемое на вентилятор. Отвечая на вопросы программы, нужно обязательно наблюдать за вентилятором (снимите крышку системного блока!) и подтвердить его остановку и запуск при указанных напряжениях. Результат калибровки *pwmconfig* записывает в файл **/etc/fancontrol**. У меня, к примеру, он выглядит так:

```
FCTEMPS=hwmon1/device/pwm1=hwmon1/device/temp1_input
FCFANS= hwmon1/device/pwm1=hwmon1/device/fan1_input
MINTEMP=hwmon1/device/pwm1=65
MAXTEMP=hwmon1/device/pwm1=85
MINSTART=hwmon1/device/pwm1=50
MINSTOP=hwmon1/device/pwm1=65
MINPWM=hwmon1/device/pwm1=65
MAXPWM=hwmon1/device/pwm1=120
```

Теперь запустим собственно **/sbin/fancontrol**, который будет использовать значение из файла выше. Смысл этих манипуляций прост: вентилятору совсем не обязательно крутиться на полных оборотах, чтобы эффективно охлаждать ЦП. При небольших загрузках ЦП вентилятор может вообще оставаться выключенным, раскручиваясь только тогда, когда температура ЦП превысит установленный лимит.

Теперь самое интересное: как отключить винчестер? Очень просто: набрав (от имени root) **hdparm -y /dev/sda**. Данная команда обесточивает жесткий диск, и тот замолкает – до первого обращения к нему. Проблема в том, что при проигрывании фильма время от времени происходит считывание файла в кэш, а это значит, что жесткий диск сразу же проснётся. Но и мы не лыком шиты: весь фильм можно кэшировать в память, если у вас её достаточно. Просто откройте фильм в *MPlayer* из консоли следующей командой: **mplayer -cache 1400000 movie.avi**. Дождитесь окончания кэширования, после чего файл **movie.avi** в текущей директории откроется в новом окне. Данная команда позволяет поместить в ОЗУ стандартный фильм в HD-качестве размером 1,37 ГБ. Перед тем как развернуть фильм на полный экран, отключите винчестер (см. выше), откиньтесь на спинку табуретки и наслаждайтесь!

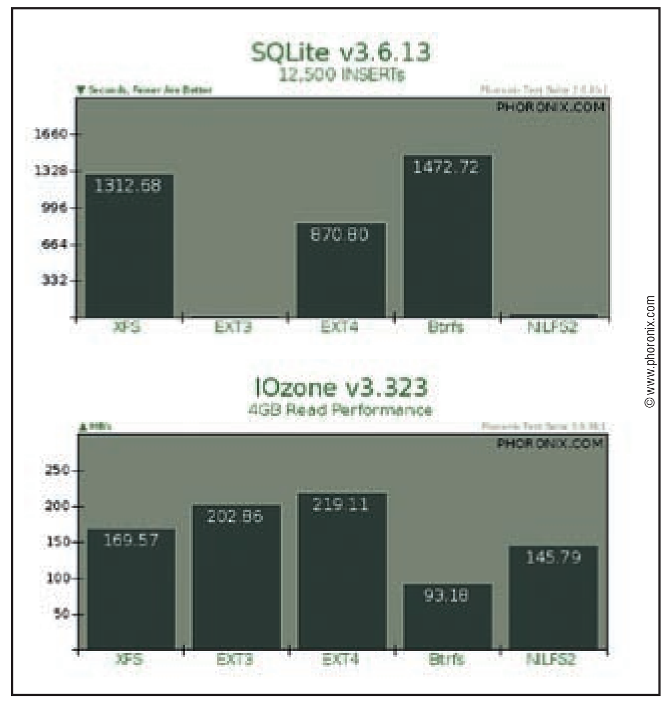
Другая файловая система

В Интернете, в местах массового скопления линуксоидов, время от времени возникают дискуссии на тему надежности файловых систем (ФС). Обсуждаются преимущества ext3/ext4 над FAT32/NTFS; файловые системы для Linux сравниваются между собой. Каждый хочет иметь одновременно и самую быструю, и самую надёжную систему, причем под надёжностью часто понимают эффективность журналирования. Сегодня мы поговорим о современной и не очень известной (пока) файловой системе NILFS, в которой реализован принцип постоянного создания контрольных точек. Другими словами, данная ФС ведет журнал всех изменений на диске в реальном времени и позволяет восстановить, к примеру, только что удаленные файлы, не прибегая к Корзине.

> «В NILFS сохранение на диск и журналирование совмещены.»

NILFS (http://www.nilfs.org) постоянно создаёт файлы журнала, при этом не перезаписывая ни один ранее созданный файл. Это очень здорово в тех ситуациях, когда возникает аппаратный сбой и нужно восстановить недозаписанные данные. С NILFS можно быть уверенным, что записанный секунду назад файл никуда не денется – ведь здесь нет никаких задержек между записью в журнал и обновлением файла на диске. Сведущие пользователи, наверное, сразу же укажут автору на ZFS из OpenSolaris, в которой реализован схожий функционал. Что же, и мы укажем на разницу между ZFS и NILFS: в OpenSolaris создание контрольной точки состояния файловой системы [snapshot] есть отдельная операция, во время которой никакая другая запись на диск производиться не может (надо подождать!). Напротив, в NILFS сохранение на диск и журналирование совмещены: любые изменения ФС выглядят как цепочка файлов, записываемых в конец раздела. Скорость чтения и записи у NILFS выше, чем у ZFS, а при решении некоторых задач (работа с БД *SQLite*) – выше в десятки раз.

Проницательный читатель знает, что за такие показатели нужно платить, и, наверное, уже догадался, что является ценой в случае с NILFS. Всё верно: постоянная запись в конец ФС приводит к её быстрому заполнению, когда формально места на диске ещё много, но служба журнала достигла физического конца раздела. Для решения этой проблемы в NILFS предусмотрен так называемый «сборщик мусора» – функция, которая время от времени удаляет совсем старые и явно ненужные журнальные записи. На месте удаленных записей получаются «дыры», и файловая система медленно, но верно фрагментируется. Еще раз уточним, что такая ситуация возникает при длительном использовании накопителя с ограниченным объёмом. На практике это означает, что при значительной фрагментации нужно делать резервную копию всех данных, форматировать раздел и записывать все данные на него заново. К счастью, необходимость в этом возникает нечасто.



› **При вставке 12,5 тысяч строк в базу данных SQLite NILFS демонстрирует отличную производительность — то же касается и чтения 4-ГБ файла.**

Для каких целей применима NILFS? Для любых, где нужна гарантия возврата к предыдущей версии файла. Также NILFS будет паллиативным решением для работы с дефектным носителем, где много «плохих» блоков. В NILFS можно подмонтировать контрольные точки параллельно с работой основной ФС и «вытянуть» данные при нестабильной работе дисковой подсистемы.

Перейдем к практике. NILFS входит в стандартную поставку ядра Linux, начиная с версии 2.6.30. Если у вас новое ядро, для начала работы нужно всего лишь установить комплект утилит для работы с файловой системой – *nilfs-utils*. Для примера возьмём подопытную карту памяти и отформатируем ее в NILFS:

```
/sbin/mkfs.nilfs2 -B 512 /dev/sdg1
```

Опция **-B** определяет количество блоков на сегмент. В данном случае пришлось понизить это значение до 512, так как карта памяти маленькая – всего 64 МБ. Стандартно NILFS работает с дисками размером от 128 МБ.

Подмонтируем раздел (**mount -t nilfs2 /dev/sdg1 /media/sdg**) и какое-то время будем сохранять на него данные. В любой момент с помощью команды *lscp* можно вывести список контрольных точек:

```
root@localhost ~]# lscp
 CNO   DATE       TIME      MODE  FLG  NBLKINC  ICNT
  1   2009-12-10  00:38:56   cp    –      11       3
  2   2009-12-10  00:39:39   cp    –      11       4
  3   2009-12-10  00:40:48   cp    –      14       5
  4   2009-12-10  00:41:53   cp    –      14       6
  5   2009-12-10  00:42:14   cp    –      14       7
  6   2009-12-10  00:42:24   cp    –      15       8
```

Первый столбец содержит числовые идентификаторы контрольных точек. Чтобы подмонтировать, к примеру, точку 2, дадим такую команду:

```
mount -t nilfs2 -r -o cp=2 /dev/sdg1 /media/cno2
```

Помимо *lscp*, пакет *nilfs-utils* включает команды для создания контрольных точек, конвертации их в снимки всей ФС и удаления. Все эти команды работают в пространстве пользователя и не требуют прав администратора.

Лифт имени Линуса

Наверное, нет такого пользователя, который не хотел бы сделать бы дисковую подсистему ещё быстрее. Один из простых и эффективных способов добиться этого – грамотный выбор планировщика ввода/вывода (I/O). Если не вдаваться в подробности, то такой планировщик упорядочивает запросы на чтение/запись в очереди ядра и позволяет сократить время поиска данных на винчестере. Как это происходит и зачем это нужно? Если бы ядро Linux обращалось к диску непосредственно при каждом системном вызове, то процесс чтения/записи превратился бы в полный хаос, и головки жёсткого диска при чтении и записи данных метались бы туда-сюда. Планировщик призван упорядочить этот процесс: он группирует и распределяет вызовы на чтение и запись данных. Планировщики бывают разными, хотя в большинстве дистрибутивов по умолчанию используется CFQ. Чтобы узнать, из чего можно выбрать, дайте команду:

```
cat /sys/block/sda/queue/scheduler
```

где sda – имя интересующего вас диска. Давайте рассмотрим особенности планировщиков.

» **CFQ** Расшифровывается как Complete Fair Queueing – полностью честная обработка очереди. Этот тип оптимизирован на множественные запросы к диску от нескольких пользователей одновременно и хорошо справляется с многозадачностью.

» **NO-OP** Самый простой вариант (no operation), обеспечивающий примитивную группировку сходных запросов. Не думайте, что он бесполезен: при интенсивной работе с USB-накопителями или с SSD-дисками, где отсутствуют затраты на перемещение головки, он может оказаться полезным, а то и здорово продлить жизнь вашим устройствам.

» **Anticipatory** Планировщик, пытающийся «угадать» следующее действие пользователя и действующий с небольшой задержкой. Его главная цель – минимизировать движение головок. Это хорошо для винчестеров, доживающих свои последние дни, но совершенно не подходит для работы с базами данных (в этом случае Anticipatory постоянно ошибается, что снижает скорость в несколько раз).

» **Deadline** Этот планировщик активно использует принцип отложенной записи. Ядро не сразу сохраняет изменения на диск, а ждет «крайнего срока», используя высвобожденное время для более быстрого считывания. Этот тип планировщика идеально годится при распределённых запросах к диску – например, при работе с БД.

Какие планировщики бывают, теперь, надеюсь, понятно; осталось разобраться, как их менять. К счастью, эта операция не требует перезагрузки, и изменения вступают в силу немедленно. Например, команда

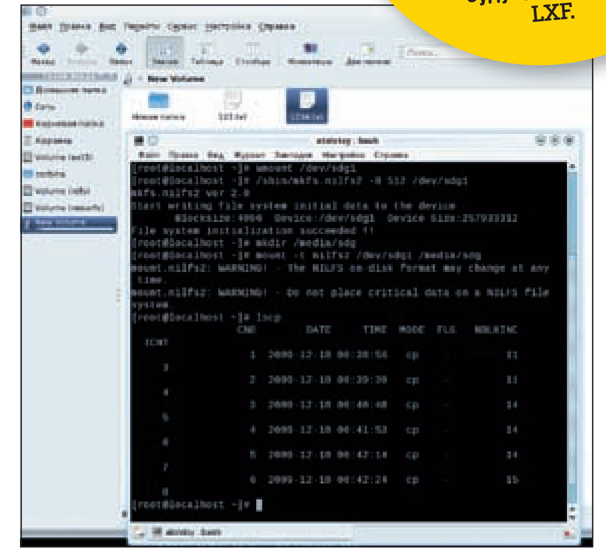
```
echo deadline > /sys/block/sda/queue/scheduler
```

набранная от имени администратора, установит тип планировщика deadline для диска **/dev/sda**.

Чтобы зафиксировать изменения на постоянной основе, следует поменять параметры, передаваемые ядру в настройках загрузчика (обычно это файл **/boot/grub/menu.lst**). В строку инициализации ядра добавьте **elevator=deadline** (или любой другой тип планировщика) и сохраните настройки. Оценку производительности диска лучше производить специальными тестами; простейший – команда **/sbin/hdparm -tT /dev/sda**. LXF

Обратная связь

Вы тоже знаете интересные хитрости и готовы поделиться ими с окружающими? Присылайте их нам на letters@linuxformat.ru – возможно, мы продолжим эту тему на страницах будущих номеров LXF.



› **NILFS — интересная файловая система, но разработчики предупреждают, что доверять ей ценные данные пока не рекомендуется.**

LilyPond

Запечатлейте свою музыку

Джон Хадсон ныряет в Tex-подобное приложение нотной записи.



Наш эксперт

Джон Хадсон начал изготовлять на компьютере документацию и раздаточный материал в 1980 г., а потом перешел на книги и *Lyx*.

Создание на компьютере нотной записи профессионального вида может превратиться в изнурительный и затратный по времени процесс. *LilyPond* [англ. «пруд с лилиями»] — продвинутая программа для набора музыки — позаботится обо всем за вас, позволяя сконцентрировать вашу энергию целиком на композиции. Как в случае с научным языком разметки *TeX*, *LilyPond* родилась в результате неудовлетворенности современными печатными материалами, но по умолчанию в ней заложена масса функций, обеспечивающих легкий старт.

LilyPond можно скачать для вашего дистрибутива с сайта http://lilypond.org, а также найти на LXFDVD. После установки, откройте текстовый редактор, введите **{e d c}** [буквенные обозначения нот ми, ре, до] и сохраните файл в виде **tmb.ly**; затем откройте консоль, перейдите в каталог, где вы сохранили **tmb.ly**, и наберите **lilypond tmb.ly**. Вы должны увидеть нечто вроде следующего вывода, а также приведенной ниже картинки:

```
GNU LilyPond 2.10.33
Processing `tbm.ly'
Parsing...
tbm.ly: 0: warning: no \version statement found, please add
\version "2.10.33"
for future compatibility
Interpreting music...
Preprocessing graphical objects...
Layout output to `tbm.ps'...
Converting to `tbm.pdf'...
```

Вы видите, что *LilyPond* создала нотный стан, добавила скрипичный ключ и размер четыре четверти и нарисовала три четвертные ноты, октавой ниже, чем нам хотелось бы. Тон первой ноты задан ниже, но соотносится с нотой до первой октавы; поэтому, чтобы начать с ми над до первой октавы, нужно либо добавить **'** к каждой ноте, либо использовать выражение **\relative**. Данные *LilyPond* указываются в фигурных скобках, внутри которых можно вводить и другие выражения, что позволяет записывать все, от отдельной строки до целой музыкальной пьесы.

› Ноты появятся относительно до первой октавы (C), поэтому сначала могут быть слишком низкими.

Длительность звучания ноты устанавливается по отношению к предыдущей (по умолчанию, длина первой ноты – четверть); поэтому для того, чтобы создать половинную третью ноту, нужно приписать **2**. Указав номер версии, упомянутой в выводе, мы получим

```
\version "2.10.33"
\relative c' {
{e d c2}
}
```

Наш PDF изменится соответственно. А теперь добавьте

```
e4 d c2 g4 f8 f e2 g4 f8 f e2
```

после c2. Нам пришлось начать с e4, потому что последняя нота была c2; без 4, обе e и d будут читаться как половинные.

Взять верную октаву

После запуска *LilyPond*, в PDF половина нот будет октавой ниже, чем надо. При использовании **\relative** тон ноты задается также относительно предыдущей: так, **g** внизу ближе ко второй **c**. Чтобы задать **g** октавой выше, нужно добавить **'** к первой **g**, но не к следующей, потому что **g** ближе к предыдущей **e**. Используйте **,** для задания ноты октавой ниже. По ходу мелодии, *LilyPond* автоматически добавит в нее тактовые черты, а также, там где надо, соединит ноты-восьмушки.

Запись нот – не единственная функция, предлагаемая *LilyPond*. Нередко требуется добавить под нотный стан слова песен. Чтобы ваш текст появился в PDF, наберите в конце документа **tbm.ly** следующий код:

```
\addlyrics {
Three blind mice, three blind mice;
see how they run, see how they run.
}
```

LilyPond автоматически припишет слово к каждой ноте; далее мы рассмотрим, как это можно изменить.

Лишь одно может показаться неинтуитивным для музыкантов: ноты вводятся на определенных линиях, а *LilyPond* затем решает, должны ли они быть с диезами или бемолями. Замените текст между **\relative** фигурными скобками на следующее:

```
\key d \major
{fis e d2 fis4 e d2 a'4 g8 g fis2 a4 g8 g fis2}
```

Диезы можно добавить с помощью **is**, а бемоли с помощью **es**. В этой ре-мажорной мелодии четыре фа-диеза, но *LilyPond* знает, что они все опускаются, поскольку перекрываются указанием ключа.

Проигрываем все вместе

Следующий приведенный код находится на DVD под названием **Cranbrook_1.ly**, вам нужно только заменить номер версии в первой строке на ваш:

```
\version "2.10.33"
\relative c' {
<<
\new ChoirStaff { \clef treble
<<
\new Voice = "Treble" { \voiceOne \relative c'
{c'2 g4. a8 g4 f e d c2. g'4 c( b) a( g) a( g) f( e) e2( d4)
s}
}
\new Voice = "Alto" { \voiceTwo \relative c'
{e2 g4. f8 e4\( d\) c\( b\) c2. e4 e( g) f( e) c2 b4( c) c2(
b4) r}
}
>>
}
\new ChoirStaff { \clef bass
<<
\new Voice = "Tenor" { \voiceThree \relative c'
{a2 d4. d8 d4\( a_~ a\) g f2. d'4 d2 d d a a2. r4}
}
\new Voice = "Bass" { \voiceFour \relative c
{d2 f4. g8 a2 a, d2. d4 d2 d g4( f) e( d) a2. s4}
}
>>
}
>>
}
```

Куски, которые нужно играть одновременно, обрамлены скобками << и >>. Таким образом, в пары скобок заключены нотный стан со скрипичным ключом и нотный стан с басовым ключом, а также партии для верхнего и нижнего голоса на каждом стане.

Начало каждого нотного стана отмечается с помощью **\new ChoirStaff**, а начало каждой партии – с помощью **\new voice**. Вы можете опустить часть голосов, но указание значений **\voiceOne** и **\voiceThree**, ноты которых имеют хвостики вверх, а также **\voiceTwo** и **\voiceFour**, у которых хвостики вниз, внесет ясность при чтении для певцов.

Ноты с точками (увеличивающими длительность) создаются – сюрприз, сюрприз! – добавлением **.** к нотам; паузы [rests] вводятся с помощью **r** и пробела, а заполняющие паузы – с помощью **s**. Там, где в нотном стане появляются паузы, нужно заполнить эти промежутки и в других партиях, чтобы *LilyPond* не занесла в свободное пространство следующую ноту мелодии.

Лиги добавляются с помощью открывающей скобки после первой ноты в лиге и закрывающей скобки после последней ноты. Во втором такте, связка в нижнем стане добавлена с помощью **~** к первой из соединяемых нот. Поскольку это верхний голос на стане, *LilyPond* по умолчанию добавляет лигу над нотами; чтобы лига добавлялась под ними, добавьте _ перед ~.



› ***Frescobaldi*, графический редактор *Lilypond*, позволяет править ваш код и просматривать PDF в том же окне.**

Лиги во втором такте являются фразовыми и введены с использованием **\(** после первой ноты и **\)** после последней ноты; такие лиги игнорируются при добавлении текста песни, тогда как обычные лиги влияют на расстановку слов. *LilyPond* автоматически добавляет место при долгих нотах и нотах с точкой для отображения их длительности, и это также создает дополнительное пространство в тех местах, где соединены несколько нот, чтобы сделать их лучше читаемыми.

Распев слов

При добавлении текста песен, как мы отметили ранее, программа по умолчанию устанавливает по слову на ноту. Можно разделить слова на несколько нот для верхнего голоса, что тут же отразится под соответствующей партией:

```
\new Lyrics \lyricsto "Treble" {
While shep -- herds watched their flocks by night,
all sea -- ted on the ground,
}
```

Использование **\addlyrics** допустимо, если понятно, какие слова идут с какими нотами; использование **\new Lyrics \lyricsto** с указанием голоса сообщает *LilyPond* о партии, под который нужно подстроить слова.

В слова **shepherd** и **seated** вставлены два дефиса, окруженных пробелами, для обозначения их растягивания на две ноты. *LilyPond* автоматически перераспределит расстояния с учетом длины слов.

Мы можем задать название и указать имя автора, добавив следующее:

```
\header {
title = "Cranbrook"
composer = "Thomas Clark 1812"
}
```

И, наконец, можно добавить рекомендованный темп, вставив следующее перед первым **\new ChoirStaff**:

```
\tempo 4 = 150
```

Религиозная и светская версии текста песни имеются на DVD: соответственно, **Cranebrook_While** и **Cranebrook_Whear**. LXF

Использование переменных

Мы могли бы обозначить каждую партию как переменную в начале файла:

```
trebleMusic = \relative c' {c'2 g4. a8 . . .}
altoMusic = \relative c' {e2 g4. f8 . . .}
tenorMusic = \relative c' {a2 d4. d8 . . .}
bassMusic = \relative c {d2 f4. g8 . . .}
```

и затем вызывать их при помощи **\trebleMusic** и так далее внутри основного кода. Таким способом гораздо проще создавать отдельные партии для певцов или музыкантов на фоне всей партитуры вашей композиции.

LINUX FORMAT на DVD
» Wireshark 1.2.2

Акула сетей

Мартин Мередит покажет, как перехватить данные, передаваемые по вашей сети, и защитить персональную информацию от разглашения.

Наш эксперт

Мартин Мередит
Разработчик Debian и Ubuntu, эксперт по безопасности крупнейшего в Великобритании розничного онлайн-магазина.

Интернет небезопасен по определению. При отправке данных через Сеть всегда существует вероятность, что кто-нибудь перехватит их, а потом доберется и до вашей персональной информации. Надеюсь, данная статья поможет вам предотвратить подобный сценарий.

Прежде чем достичь адресата, данные проходят через множество соединений, путешествуя по Интернету. Многие даже не догадываются, что по дороге эти сведения могут быть считаны на любой машине, входящей в цепочку.

Имея подходящие инструменты, вы можете сами перехватывать эти – да и любые другие – данные, циркулирующие в сети. Это возможно потому, что большинство сетей фактически рассылают информацию, предназначенную для конкретного адресата, всем сетевым машинам подряд; просто ваш компьютер игнорирует чужие пакеты. В особенности это справедливо для большинства беспроводных сетей, даже «защищенных» с помощью WEP/WPA.



› *Wireshark* позволяет прослушивать любое сетевое подключение.

Ловись, рыбка...

Давайте попробуем перехватить данные в вашей сети. Сначала установим инструмент под названием *Wireshark*. В большинстве дистрибутивов программу можно получить с помощью менеджера пакетов; в Ubuntu я набираю следующую команду:

```
sudo apt-get install wireshark
```

Если в вашем дистрибутиве такого пакета не нашлось, исходный код можно найти на www.wireshark.org/download.html – вместе с инструкциями по компиляции и установке.

Обычно *Wireshark* подчиняется только пользователю root. Чтобы запустить программу, введите команду **gksu wireshark** или аналогичную. У вас спросят пароль суперпользователя, после чего *Wireshark* запустится (как обычно, уведомив об опасности работы в режиме root). В рамках нашего руководства это предупреждение можно игнорировать, но запуск любой программы от имени суперпользователя опасен, поэтому будьте осторожны!

После старта *Wireshark* вы увидите пустой экран с несколькими значками вверху. До поры мы не будем вникать в эти детали, поскольку сначала нужно перехватить какие-нибудь данные. В *Wireshark* перейдите к меню Capture [Захват] и выберите

пункт Interfaces [Интерфейсы]. В появившемся диалоге мы выберем интерфейсы, с которых будут считываться данные. Проще всего указать Any [Любые], что на языке *Wireshark* означает «собрать все данные, которые я только вижу в сети». Нажмите кнопку Start [Пуск] рядом с интерфейсом Any – и вы увидите главный экран, разделенный на три части. Верхняя часть начнет заполняться данными. Если вы, как и я, единственный пользователь сети (или ваша сеть построена на переключателях), то сведений будет немного. Пошарьте по Интернету, и данных станет больше (*Wireshark* захватывает любую информацию, поступающую на ваш компьютер и отправляемую с него – и вообще все, что попадется в сети).

Перехват WordPress

Чтобы привести пример данных, подверженных перехвату, я выполнил локальную инсталляцию *WordPress* и собрал данные, сгенерированные в ходе установки.

Я знаю, что *WordPress* – это web-приложение, а значит, использует протокол HTTP; поэтому я предписал *Wireshark* искать только HTTP-пакеты – а именно, просто ввел **http** в поле фильтра. После этого начинается заполнение информационной колонки данными. Давайте просмотрим, как я вошел в *WordPress*.

Отправляя форму через Интернет, вы посылаете данные на сервер с помощью метода HTTP под названием **POST**. Запрашивая страницу, вы пользуетесь методом **GET** этого же протокола. Чтобы засечь вход в панель администрирования *WordPress*, мне нужно отыскать запрос **POST**, отправленный сценарию входа в систему. *Wireshark* предоставляет множество способов манипулирования данными: давайте просмотрим все сделанные мной запросы **POST**.

Нажмите кнопку Clear [Очистить] рядом с полем фильтра, затем кнопку Expression [Выражение]. В окне можно составить формулу, чтобы отфильтровать данные. Глянув в список Field Name [Название поля], можно заметить, сколько всего понимает *Wireshark*, но пока нас интересуют только запросы **POST**, поэтому прокрутим список вниз до пункта HTTP и развернем его, щелкнув стрелку. В развернувшемся списке найдем пункт **http.request.method** – то есть будем отсеивать данные на основании метода запроса. Выберите пункт **http.request.method**, в поле Relation [Отношение] укажите **==**, а в качестве Value [Значение] введите **POST**; затем нажмите OK.

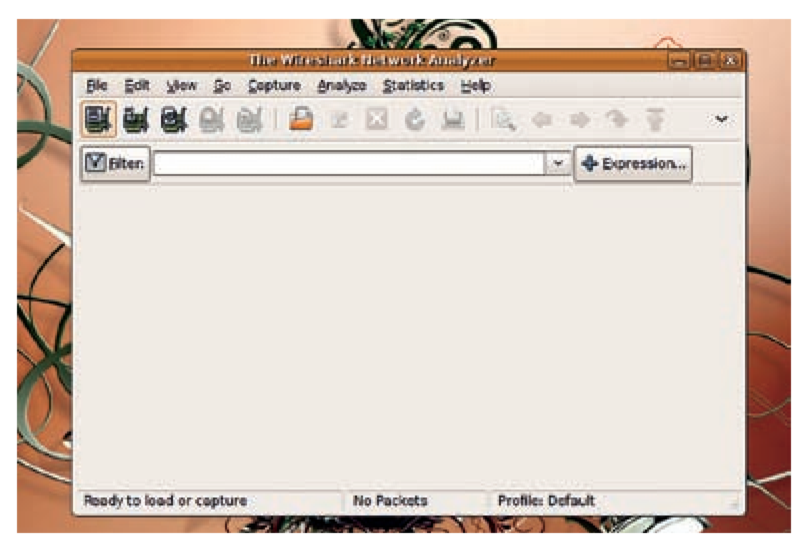
Список захваченных пакетов сократится до нескольких пунктов, и среди них будет легко отыскать тот самый, соответствующий отправленному мной запросу к **/wordpress/wp-login.php**. Давайте рассмотрим его подробнее.

Вы заметите, что экран теперь разделен на три части (если нет, попробуйте развернуть окно). Центральная часть окна позволяет изучить пакет во всех деталях. Нас интересует раздел Line-based text data [Текстовые строки], в котором всего одна строка, вот такая:

```
log=admin&pwd=HCnr9%5E%40rWsbt&wp-submit=Log+In&redirect_to=http%3A%2F%2Flocalhost%2Fwordpress%2Fwp-admin%2F&testcookie=1
```

Перехват данных WordPress

Параметр	Значение
log	admin
pwd	HCnr9^@Wsbt
wp-submit	Log in
redirect-to	http://localhost/wordpress/wp-admin
testcookie	1



› Аскетический интерфейс вскоре заполнится данными всех сортов...

Эти данные представляют собой пары «ключ–значение», закодированные по схеме URL. Декодируем их – получим сведения, приведенные в таблице *внизу слева*.

Узнаете? Это имя пользователя и пароль, с которыми вы вошли в *WordPress*. То есть тот, кто перехватил ваши данные, получил доступ к вашему блогу и теперь может вытворять там все, что ему вздумается, пока вы не смените пароль.

Какой ужас!

А что с электронной почтой? Если у вас обычный почтовый клиент, то вы можете перехватить переговоры вроде этих:

```
< 220 stupor.sourceguru.net ESMTP Postfix (Ubuntu)
> EHLO sourceguru.net
< 250-stupor.sourceguru.net
> AUTH PLAIN JiM2NTUzMztseGZ1c2VyJiM2NTUzMztse
GZwYXNzd29yZA
< 235 2.7.0 Authentication successful
> MAIL FROM:<mez@sourceguru.net> SIZE=456 AUTH=<>
< 250 2.1.0 Ok
> RCPT TO:<mez@debian.org> ORCPT=rfc822;mez@debian.
org
< 250 2.1.0 Ok
> DATA
< 354 End data with <CR><LF>.<CR><LF>
> Date: Sun, 13 Sep 2009 20:59:06 +0100
> From: Martin Meredith <mez@debian.org>
> To: mez@debian.org
```

»

Скорая помощь

Если вы пользуетесь Google Mail, перейдите в меню Настройки и установите переключатель Использовать только HTTPS, чтобы всегда подключаться только через безопасное соединение.

Сертификаты SSL

Как правило, сертификат SSL выдается центром сертификации как подтверждение личности предъявителя. Если адресная строка в вашем браузере окрашена в зеленый цвет, то сертификат, отправленный вам, прошел Расширенную проверку [Extended Verification]. Значит, вы можете быть уверены в том, что сервер, к которому вы обращаетесь, принадлежит именно той компании, с которой вы ведете переговоры.

Большинство центров берут плату за подтверждение вашей личности, и в некоторых случаях она доходит до тысяч долларов. Альтернативный способ для тех, у кого нет лишних денег на покупку коммерческого сертификата – обращение в CAcert. Это организация, целью которой является выдача бесплатных подписанных сертификатов всем пользователям. Подтверждение личности пользователей в CAcert осуществляется с помощью модели Web of Trust (WoT), то есть «сети доверия». Чтобы стать подтвержденным пользователем CAcert, нужно отыскать других пользователей этой сети и предъявить им доказательства подлинности вашей личности. Эти пользователи (если доказательства их устроят) начисляют вам «очки». Набрав 50 очков, вы сможете создать «подтвержденный» сертификат, то есть CAcert в этом случае будет уверена в вашей личности. Технология CAcert распознается пока не всеми браузерами, но большинство дистрибутивов Linux снабжается файлами, позволяющими пользоваться CAcert-сертификатами наравне с коммерческими.

```
> Subject: Test Email
>
> Wireshark can sniff your email too
> .
¬o< 250 2.0.0 Ok: queued as 095CA94024
< QUIT
< 221 2.0.0 Bye
```

Это пострашнее, чем в предыдущем примере. Вообразите, что кто-то сидит около вашего дома с ноутбуком и прослушивает трафик Wi-Fi [если сеть не зашифрована или злоумышленник узнал ключ: для WEP это несложно, – *прим. ред.*]. Он может не только прочесть вашу почту, но, из безобидной строчки с началом **AUTH PLAIN** выяснить имя пользователя и пароль вашей учетной записи (для нашего примера я заменил их на **lxfuser** и **lxfpassword**). Метод авторизации **AUTH PLAIN** попросту объединяет ваши имя пользователя и пароль и кодирует их посредством **base64** (то есть расшифровать сообщение может любой – загляните хоть на http://linkpot.net/scripture).

Как же пресечь шпионаж?

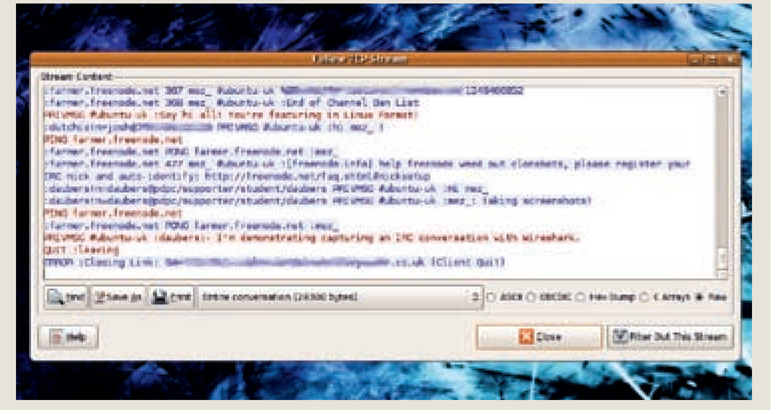
Вышеприведенные примеры показывают, как просто получить секретные данные, прослушивая канал. Лучший способ предотвратить это – закодировать данные, то есть отправить их в виде, исключающем декодирование посторонними. Иначе этот способ называется шифрованием. Если вы когда-либо делали покупки онлайн, то, вероятно, заметили непривычно зеленую (реже – синюю) окраску адресной строки браузера при совершении платежа. Это визуальное подтверждение того факта, что ваш браузер использует безопасное соединение.

> «Для шифрования чаще всего применяются методы SSL и TLS.»

Чаще всего используются методы шифрования SSL (Secure Sockets Layer) и TLS (Transport Layer Security). Оба метода применяют комбинацию сертификатов и ключей. Клиент (то есть вы) получает сертификат от сервера и использует его для шифрования данных, отправляемых на сервер. Расшифровать их можно только с помощью приватных ключей, находящихся на сервере. Приватный ключ, в свою очередь, зашифровывает данные так, что расшифровать их можно только открытым ключом, встроенным в сертификат. Для усиления безопасности в момент соединения генерируется случайный ключ, который, совместно с приватным ключом и сертификатом, используется для шифрования обмена данными между вами и сервером.

Слежение за переговорами

Иногда *Wireshark* прямо-таки заваливает вас данными, особенно при ловле их в сети с бойким трафиком (скажем, у вас в офисе). Чтобы облегчить вам жизнь, *Wireshark* предусмотрел функцию отслеживания потока TCP: она сводит воедино все пакеты, касающиеся обмена между двумя машинами. Вообразите, например, что вы, прочесав данные перехвата, заметили пакеты чьих-то переговоров по IRC. Чтобы за ними проследить, достаточно нажать правую кнопку мыши и выбрать команду Follow TCP Stream [Отслеживать поток TCP].



› ***Wireshark*** **умеет следить за переговорами по протоколу TCP — здесь показано, как #ubuntu-uk болтает через IRC.**



› **Зеленый цвет адресной строки показывает, что соединение с сервером шифрованное.**

Чтобы иметь возможность устанавливать защищенное соединение самому, необходимо получить SSL-сертификат для вашего сервера. Сначала сгенерируем собственный приватный ключ. Для этого наберите команду:

```
openssl -genkey -out privey.pem 4096
```

Приватный ключ будет сгенерирован с помощью системной энтропии. Создав ключ, вы должны сгенерировать запрос на подпись сертификата (Certificate Signing Request, CSR). Как правило, SSL-сертификаты проверяются центром сертификации (Certificate Authority, CA), который удостоверяет вашу личность и подписывает сертификат. Запрос CSR можно создать, набрав следующую команду:

```
openssl req -new -key privkey.pem -out cert.csr
```

После этого вам будет предложено ввести некоторые детали. Укажите их и проследите, чтобы параметр CommonName (CN) соответствовал доменному имени вашего сервера:

```
mez@lazy % openssl req -new -key privkey.pem -out cert.csr
Country Name (2 letter code) [AU]:UK
State or Province Name (full name) [Some-State]:West
Midlands
Locality Name (eg, city) []:Birmingham
Organization Name (eg, company) [Internet Widgits Pty Ltd]:.
Organizational Unit Name (eg, section) []:.
Common Name (eg, YOUR name) []:lazy.sourceguru.net
Email Address []:martin@sourceguru.net
```

Теперь нужно направить запрос CSR в центр сертификации, который взамен вышлет подписанный сертификат. Мы сохраним его под именем **signed.pem**. Для упрощения задачи объединим ключ и сертификат в одном файле, чтобы указать на него при настройке.

```
cat privkey.pem signed.pem > certificate.pem
```

Ну вот, у нас есть ключ и сертификат SSL. Попробуем применить их на практике.

Настройка серверов

Чаще всего SSL-сертификаты используются для обеспечения безопасности сайтов, поэтому неудивительно, что в *Apache* процесс настройки защищенного сервера очень прост. Во-первых,

Случайно == безопаснее

Во многих методах шифрования для обеспечения дополнительной безопасности используется элемент случайности. На массе машин (например, настольных Linux-компьютерах, VPS и т.п.) случайные данные (называемые энтропией) вырабатываются постоянно. В Linux-системах энтропия основывается на измерении различных параметров: например, микроскопической временной разницы между обращениями к жесткому диску. Если ваше приложение нечасто обращается к винчестеру, то в системе может оказаться недостаточно энтропии, и злоумышленник сможет «угадать» случайные числа.

Компания Simtec Electronics выпускает продукт, который решит проблему: это маленький USB-накопитель, подключаемый к USB-порту. С помощью генераторов квантового шума, математических и криптографических манипуляций он может поставлять вашей системе случайные числа в больших количествах. Более подробные сведения можно найти на www.entropykey.com.





› **Для запуска *Wireshark* нужны привилегии суперпользователя, поэтому будьте осторожны, не испортите что-нибудь ненароком!**

мы должны организовать «сайт» внутри *Apache*. Сделать это можно, создав файл в каталоге **/etc/apache2/sites-available** (путь к файлу будет такой: **/etc/apache2/sites-available/lazy.sourceguru.net**). Содержание файла:

```
NameVirtualHost *:443
<VirtualHost *:443>
  ServerName lazy.sourceguru.net
  DocumentRoot /var/www/
  SSLEngine On
  SSLCertificateFile /путь/к/certificate.pem
</VirtualHost>
```

Проделав это, необходимо активировать SSL-модуль в *Apache*, подключить только что созданный файл и перезапустить *Apache*:

```
a2enmod ssl
a2ensite lazy.sourceguru.net
/etc/init.d/apache2 restart
```

Если теперь вы попытаетесь посетить свой сайт, например, по адресу https://lazy.sourceguru.net, то увидите, что подключение зашифровано (адресная строка окрашена в зеленый или синий цвет).

С почтой несколько сложнее, поскольку в большинстве случаев у вас будет один сервер для получения, а другой – для отправки писем. В моем примере использованы *Postfix* и *Courier*, но даже если у вас будет другая конфигурация, в Интернете вы найдете массу руководств по данному вопросу.

Безопасная почта

Сперва настроим *Postfix*. Если конфигурация вашей почты требует использования доменного имени, отличного от использованного выше, понадобится получить другой сертификат. После этого настроить *Postfix* не составит труда. Добавьте в файл **/etc/postfix/main.cf** следующие строки:

```
smtpd_tls_cert_file = /path/to/certificate.pem
smtpd_tls_key_file = $smtpd_tls_cert_file
smtpd_tls_log_level = 0
smtpd_tls_recieved_header = yes
smtpd_tls_security_level = may
smtpd_use_tls = yes
smtpd_tls_auth_only = yes
```

Перезапустите *Postfix*, и вы сможете использовать TLS.

Поддержка POP и IMAP обеспечивается службой *Courier*. Здесь мы рассмотрим процесс настройки POP, но для IMAP шаги будут почти такие же.

В Ubuntu первым делом нужно установить пакет *courier-pop-ssl*. Закончив установку, скопируйте файл сертификата в **/etc/courier/pop3d.pem**. Затем отредактируйте **/etc/courier/pop3d**, заменив строку

```
POP3DSTART=YES
```

на

```
POP3DSTART=NO
```

Это отменяет запуск стандартных интерфейсов POP3. Но так как они, вероятно, уже работают, необходимо остановить их и перезапустить экземпляр SSL:

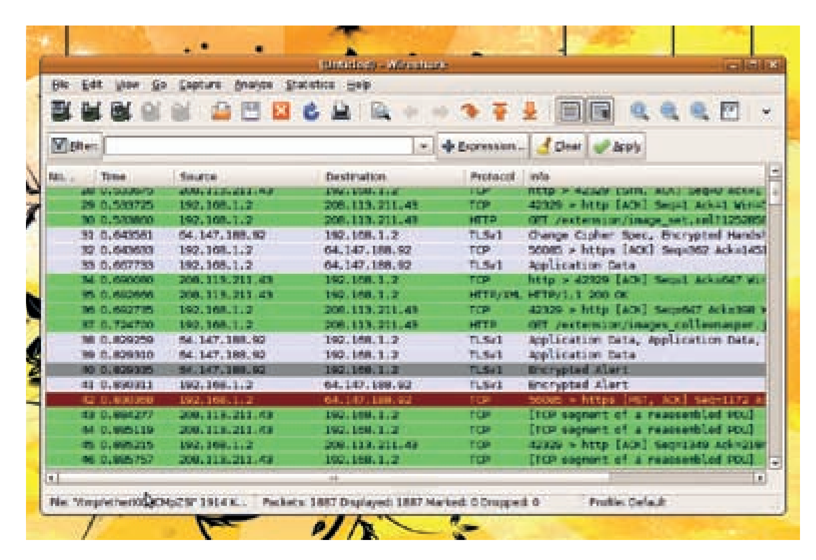
```
/etc/init.d/courier-pop stop
/etc/init.d/courier-pop-ssl restart
```

Теперь вы обезопасили свою почту. Не забудьте обновить настройки почтового клиента!

Как мы отметили в начале статьи, Интернет небезопасен по определению. Простейший и эффективнейший способ бороться с этим – шифрование. Мы показали вам, как шифровать входящий и исходящий трафик, но это далеко не все. Любой протокол, используемый в Интернете, подвержен перехвату – от службы мгновенных сообщений до пикантных видеороликов, которые вы смотрите в три часа пополуночи. Поэтому используйте шифрование где только возможно: большинство серверов позволяют настроить такое соединение просто и быстро. Если вы программист и пишете новую программу – подумайте о пользователях и обеспечьте им возможность безопасного обращения к вашим услугам. LXF

Скорая помощь

Чтобы программа автоматически прекратила захват в определенное время, воспользуйтесь меню Capture > Options [Захват > Параметры].



› ***Wireshark* перехватывает уйму данных, но далеко не все они полезны.**

Что за штука... Maemo?

Боб Мосс разведал планы Nokia по нейтрализации iPhone. Секретным оружием послужит ОС на базе Linux с полным боекомплектом функций.

Полагаю, название не имеет отношения к деревеньке в Тибете?
Справедливо полагаете! Мы говорим об операционных системах на основе Linux, предназначенных для смартфонов Nokia. Самый яркий пример такого смартфона – N900, мы рассмотрим его позже. В произношении уверенности не было, и мы расспросили членов команды разработчиков. Сошлись на том, что название должно звучать как «мэй-мо».

Ах, Maemo основан на Linux? С этого места, пожалуйста, подробнее!
Пользовательский интерфейс в Maemo построен на базе каркаса Hildon. Описывая *Clutter* в **LXF123**, мы упоминали, что Hildon – это набор расширений *GTK* для графического интерфейса Gnome Mobile. Однако, в отличие от Moblin, вы не найдете *Clutter* на своем Nokia-смартфоне! Текущая версия Maemo – 5; но прошлая, OS2008, под кодовым названием «Diablo», также используется на интернет-планшетах Nokia предыдущего поколения.

> «Можно переговариваться по Skype через сеть сотовой связи.»

При такой поддержке *GTK* для Maemo должно быть доступно немало Gnome-приложений, не так ли?
Совершенно верно; самые заметные – *Pidgin*, *VLC* и VNC-клиент. Можно загрузить родные для Maemo приложения, которые добавляют поддержку OGG или обеспечивают функциональность, подобную iPhone.

По-моему, Nokia приобрела компанию rolltech, создателя *Qt* — а где же KDE-приложения?
Trolltech и в самом деле куплена, но *Qt4* не будет применяться для разработки интерфейса до выхода OS2010 (кодовое имя «Harmattan»). Собственно, создавать приложения для Maemo с помощью *Qt4* можно уже сейчас, если загрузить соответствующий пакет совместимости. Ходят слухи, что Nokia откладывает использование *Qt4* до той поры, пока приложения KDE 4 не станут более стабильными, но истинные причины нам еще предстоит узнать.

Хорошо... А правда, что Maemo выделывает те же трюки, что и iPhone?
Не только те же, но и посложнее. Обеспечив копирование со вставкой, календари и управление контактами и фотографиями, Maemo идет дальше, интегрируя контакты из источников типа *Google Talk* и *Skype* и обеспечивая упрощенный доступ к ним. Отображается новейшая аватара контакта и его состояние (в сети или нет).

Действительно круто! И стандартные приложения тоже лучше?
Календарь Maemo гораздо интуитивнее и более эффективно использует экранное пространство; подобно пользователям iPhone, вы можете свободно синхронизировать календари и контакты через серверы Microsoft Exchange. Браузер Maemo отображает Flash-контент и поддерживает полноэкранный серфинг, поскольку базируется на одном движке (*Gecko*) с *Mozilla Firefox*. По сути, это означает, что можно просматривать любой стандартный сайт, доступный с настольного Linux-компьютера.

Что скажете о камере?
У Nokia N900 5-мегапиксельная камера, с помощью которой Maemo умеет записывать видео высокого разрешения. Режим съемки автоматически подстраивается под окружающие условия; кадры отмечаются тэгами. Можно выгружать фото и видео на онлайн-сервисы типа Facebook или Flickr, или на собственный сайт Nokia под названием Ovi. Чтобы проделать все это на iPhone, вам понадобится несколько разных приложений. И даже если удастся выпустить их из «песочницы» (а в Британии это незаконно), то высокое разрешение будет все же недоступно.

Мне на мобильном телефоне нужна карта. Есть ли она в Maemo?
Конечно, есть. Maemo полноценно использует GPS-чип N900. Определение текущего местоположения и приемы работы с картой реализованы примерно так же, как на iPhone. При этом на картах Ovi обозначаются важные объекты и локализованные сводки погоды. Больше того, возможно объемное отображение некоторых достопримечательностей (правда, пока это скорее курьез, чем серьезная инновация). Опять же, сокращается количество приложений для выполнения одной задачи.

А можно прослушивать с помощью смартфона N900 музыкальные коллекции *iTunes*? У меня там богатая подборка...
Стандартный медиа-плейер Maemo поддерживает MP4 и AAC, но прослушать DRM-защищенные материалы, купленные в iTunes Store, вы не сможете. Это хорошо документированная проблема Linux. Но если контент без DRM-защиты был конвертирован в формат Ogg Vorbis, то можно установить необходимые кодеки и прослушивать музыку из коллекции по официальным каналам.

Ну, а что в Maemo такого, чтобы реально захотелось переключиться?
У вас появится гораздо больше свободы для хакинга вашего устройства; кроме того, можно будет избежать многих ограничений. Например, с помощью смартфона Nokia можно организовать бесплатные переговоры со *Skype* на *Skype* че-



рез сеть сотовой связи. Причем взаимодействие с собеседником не ограничивается одними разговорами: дискуссия, начатая в текстовом режиме, может перейти в *Skype*-беседу, а завершиться чатом в службе мгновенных сообщений – текст позволяет хорошенько обдумать ситуацию и вернуться к переговорам позже.

Хитро придумано! Но где же убойные функции?

Обновления системы Maemo выполняются автоматически: незачем по старинке проверять их наличие и устанавливать вручную. Вдобавок вы будете получать уведомления о появившихся обновлениях для программ, установленных вами с помощью Maemo Select (собственного хранилища приложений Nokia). Здесь уж вам решать, устанавливать обновления или нет. Можно также запускать несколько приложений одновременно, переключаться между ними и даже расположить окна мозаикой для наблюдения за ходом работы. Уж этого-то вы нигде больше не получите! А поскольку вы имеете дело с Linux, у вас будет столько рабочих столов, на сколько вам хватит мыслительных способностей, и каждый рабочий стол можно настроить отдельно.

Вау! Действительно впечатляет. А много ли доступно приложений?

На момент подписания номера в печать в Maemo Select было только 50 программ, но это не удивительно: речь идет о системе, первые устройства под управлением которой только-только появились на рынке. Это означает, что в сообществе Maemo, которое имеет тенденцию к росту, будут проявлять особую заботу о разработчиках открытого ПО.

Какие условия созданы для их работы?

Существует SDK Maemo, рассчитанный на программистов-линуксоидов. Есть привязки к C, Java, Python, Ruby и Mono; благодаря каркасу Hildon разработчикам, использующим для создания графических интерфейсов *GTK*, доступна мощная поддержка. С помощью сервиса Maemo Garage можно размещать проект онлайн.

Расскажите-ка о Maemo Garage...

Это бесплатное серверное пространство, где можно хранить сообщения о найденных ошибках и состоянии проекта, а также организовать список рассылки для поддержки пользователей. Здесь же можно хранить исходный код и координировать совместные действия с членами своего сообщества. Преимущество размещения проекта в «гараже» заключается в том, что здесь легче найти помощника с достаточным уровнем знаний.

А SDK будет помещать закрытые компоненты в мой свободный код?

Организаторы сообщества Maemo предусмотрели выбор: во время инсталляции SDK вы сами решаете, устанавливать несвободные компоненты или нет. Это вполне в духе отношения Nokia к свободному ПО.

Уговорили! Пойду куплю Nokia N900! Где узнать горячие новости о Maemo?

Последние известия можно получить на http://maemo.nokia.com, там есть также ссылки для покупки N900. А чтобы поучаствовать в разработке приложений, направляйтесь на http://maemo.org. LXF



По рецептам доктора Брауна

Д-р Крис Браун
Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

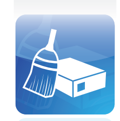
Добротное администрирование систем из причудливых заворотов кишок серверной.













Похоронный блюз



Переходя от возвышенного к нелепому (или наоборот), я предлагаю вам небольшой стишок, с искренними извинениями У.Х. Одену [W.H. Auden, пародируемый здесь поэт]:

Часы отключите, обесточьте ядро!
Мой верный ноутбук обратился в ничто…
И кабель, и мышь извлеките навек,
Поставив на траурный марш саундтрек.

Избавьте от пошлых сочувственных сцен –
Нетбук, Inspiron не вручайте взамен;
Я в курсе, что техника мчится вперед,
Но счастья быть вместе никто не вернет.

Он был моим западом, севером, югом…
Я думал, лет пять еще будем друг с другом.
Я думал: «Давно ль я его запустил?».
Я думал, диск вечен.
Неправ же я был!

Пора слать по почте трагичную весть,
Бэкапы с кассет искать способ прочесть,
Под дробь барабана крышку закрыть
И экологов-мусорщиков впустить.

(Чтобы найти оригинал, наберите «stop all the clocks» в Google.)

Освежающе легко

Eucaliptus Создайте свое облако с помощью пары Кармических Коал и веточки Эвкалипта.

Если вам нравится концепция «эластичных облачных вычислений» или «самообслуживаемой ИТ-инфраструктуры», но вы не решаетесь доверить вычислительные потребности целой организации кому-то со стороны, воспользуйтесь новым предложением Canonical – Ubuntu Enterprise Cloud (UEC – Облако Ubuntu для предприятий). Оно позволяет создать частное облако типа Amazon EC2 в локальной сети. В основе UEC лежит Eucaliptus, открытая реализация развивающегося де-факто стандарта API EC2. Eucaliptus был разработан в UCSB (Университет Калифорнии в Санта-Барбара), а название, как нетрудно догадаться, является сокращением от Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems [Эластичная вычислительная компьютерная архитектура для связи ваших программ с полезными системами]. Именно за это название Регенты Университета Калифорнии запретили ребятам из Санта-Барбара давать имена чему-либо до 2012 года (а если не запретили, то следовало бы...).

У Canonical есть официальная документация по продукту – *Ubuntu Enterprise Cloud Architecture* (зайдите на www.ubuntu.com и кликните на Server Edition, а потом на Cloud Computing). В ней они пишут: «В следующих релизах Ubuntu наша цель – создать полноценный слой инфраструктуры для облака, чтобы каждый желающий мог развернуть его для личного (частное облако) или общего пользования». Вообще говоря, Eucaliptus появился в Universe-репозитории Ubuntu еще в версии 9.04 (Jaunty), но вышел на первый план лишь в серверной версии 9.10 (Karmic Koala). Я уверен, что связь коалы с эвкалиптом – не просто совпадение.



› **Видеозапись рассказа Саймона Уордли [Simon Wardley] на OSCON 2009 об облачных вычислениях есть на tinyurl.com/mo28ol.**

UEC – один из трех компонентов «облачной» стратегии Canonical. Остальные два – Ubuntu One (сервис синхронизации, разделения и резервного копирования файлов, о котором я рассказывал в **LXF124**) и образы Ubuntu Server Edition, которые Canonical сделала доступными в облаке Amazon EC2. Хотя UEC и Eucaliptus построены по модели Amazon EC2 API, в реализации есть ключевое различие. Amazon использует технологию виртуализации *Xen*, а Canonical – *KVM*. Но я подозреваю, что большинство пользователей не знают или не думают об этом.

Вы можете узнать о *Xen* больше из урока по Live CD на смежной странице или открыв стр. 52 с рассказом о моих упражнениях с облаком Amazon.

Первые шаги с Xen

Познакомимся с **Xen**, загрузив Live CD и загрузившись с него.

Xen – популярная технология виртуализации. Она началась как исследовательский проект в компьютерной лаборатории Кембриджского университета, а затем была взращена и поддержана XenSource Inc (ныне принадлежит Citrix). *Xen* – это гипервизор, программный слой, расположенный ниже ОС и работающий вместе с привилегированным экземпляром ОС под названием «Domain 0» или просто «Dom0» (домен в терминах *Xen* – просто экземпляр виртуальной машины, ВМ). При загрузке системы на основе *Xen* сначала стартует гипервизор, и машина с Dom0 запускается автоматически. Гипервизор и Dom0 вместе управляют аппаратными ресурсами и поддерживают создание дополнительных ВМ – «пользовательских доменов», или «DomU».

Xen запустится на 32- или 64-битном компьютере с поддержкой виртуализации процессором. На компьютерах без аппаратной поддержки виртуализации Xen предоставляет паравиртуализацию; в этом случае ядра компьютеров Dom0 и DomU нужно модифицировать, заменив ключевые моменты программными прерываниями (гипервызовами) гипервизора. К счастью, *Xen*-совместимые ядра Linux найти нетрудно. На компьютерах с поддержкой аппаратной виртуализации (т. е. процессорах с расширениями Intel-VT или AMD-V) *Xen* предоставляет полную виртуализацию и позволяет запускать немодифицированные ОС, и можно установить в DomU оригинальные версии Linux или даже Windows.

Я решил познакомься с *Xen*, взяв образ диска на http://wiki.xensource.com/xenwiki. Следуйте за мной: будет интересно.

Момент Xen

Загрузочное меню диска предлагает запустить *Xen* с Dom0 на основе Debian (Lenny) или просто стартовать Lenny. Как и любому Live CD, ему требуется время на загрузку, но вскоре шум привода затихает, и появляется рабочий стол Gnome с запущенным *Virt-manager*. (*Virt-manager* – графическая утилита управления ВМ на основе *Xen*, *KVM* или других технологий виртуализации; см. **LXF125**). Оказалось, что на диске четыре DomU – два сервера и два клиента рабочего стола, все на основе Ubuntu 8.10, но только два из них запускаются автоматически.

Я предпочел обойтись без *Virt-manager* и воспользовался *xm*, стандартной командой Xen для взаимодействия с доменами. Например, вывести список доменов можно таким образом:

```
# xm list
Name       ID   Mem    VCPUs   State    Time(s)
Domain-0   0    640    2       r-----   55.1
client01   2    256    1       -b----   3.2
client02        256    1                0.0
server01   1    64     1       -b----   5.1
```

а загрузить второй серверный домен – так:

```
# xm start server02 -c
```

Обратите внимание на флаг **-c**: он отвечает за немедленное подключение к консоли запущенного компьютера, и благодаря ему я вижу сообщения, выводимые во время загрузки. Подключаться к консолям своих доменов или отключиться от них можно и по желанию. Для подключения и входа в систему, выполните команды:

```
# xm console server01
Ubuntu 8.10 server01 hvc0
server01 login: livexen
Password:
Last login: Sat Oct 17 03:07:18 BRT 2009 on tty1
livexen@server01:~$
```

Для отключения нажмите Ctrl+]. «Уничтожается» домен так:

```
# xm destroy server02
```

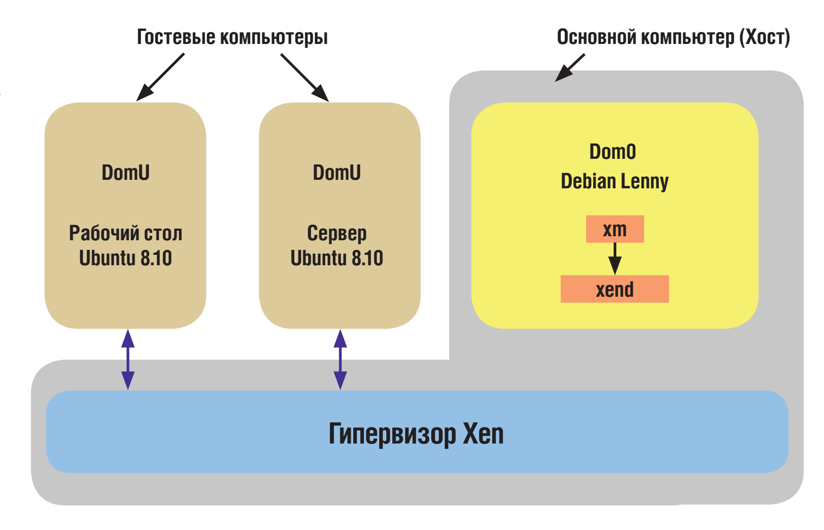
Эта команда эквивалентна выдергиванию шнура настоящего компьютера из розетки.

Ради углубления в тему, вот несколько упражнений на пробу:

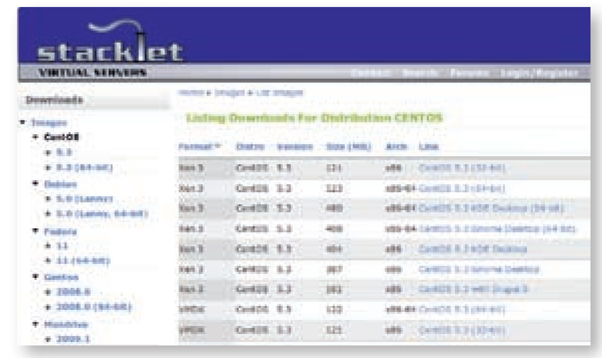
» Просмотрите файлы конфигурации в **/etc/xen**. Они определяют виртуальное оборудование для систем, запущенных в DomU.

» Поиграйте с *Virt-manager*. Он дает альтернативную возможность запуска/останова доменов, а также подключения к графическому рабочему столу клиентских машин в DomU.

» Установите в один из клиентских DomU пакет *openssh-server* и соединитесь с ним из Dom0 по SSH.



› **Архитектура *Xen* на Live CD. *xm*, родная утилита управления виртуальными машинами в *Xen*, подключается к гипервизору через демон *xend*.**



› **На http://stacklet.com можно найти образы многих дистрибутивов Linux, подготовленные для *Xen*, *Qemu* или в формате VMDK.**

Книга о виртуализации с Xen

Я теперь выучился лучше держать в узде свою страсть к покупке книг, но все же не утерпел и приобрел «Professional Xen Virtualization» (Профессиональная виртуализация с Xen) Уильяма фон Хагена [William von Hagen], от издательства Wrox Press (www.wrox.com).

Книга начинается с заслуживающего доверия обзора технологий виртуализации и существующих продуктов, вводящего в повествование о *Xen*. В последующих главах раскрываются вопросы установки *Xen*, запуска и настройки хоста *Xen*, создания и конфигурирования ВМ и настройки сети.

Эта книга добросовестно и аккуратно написана компетентным автором, и я счастлив вам ее посоветовать.

В будущее на облаке

Узнаем, как арендовать кусочек облака Amazon.

Под впечатлением от экспериментов с *Xen* я завел учетную запись на Amazon Web Services (AWS). Amazon предлагает множество сервисов, включающих:

» **EC2 (Elastic Computing Cloud)** Предоставляет вычислительные мощности, размер которых способен меняться за счет быстрого создания экземпляров виртуальных машин. На выбор предлагается множество сконфигурированных образов виртуальных машин и несколько вариантов оборудования.

» **E3C (Simple Storage Service)** Объектно-ориентированное интернет-хранилище.

» **EBS (Elastic Block Store)** Предоставляет блочное устройство (в терминах ядра Linux) для размещения файловой системы, подключаемое к любой ВМ. Тома EBS не зависят от запущенных экземпляров ВМ.

» **Cloudwatch** Служба мониторинга, который строит графики потребления процессора, трафика и дискового пространства экземпляром ВМ.

» **Cloudfront** Сервис быстрой доставки web-содержимого через глобальную сеть узлов – выбирается ближайший к получателю.

» **Elastic IPs** Позволяет иметь стабильные публичные IP-адреса и назначать их на любой экземпляр системы под вашей учетной записью.

» **SQS (Simple Queue Service)** Предоставляет надежную, хорошо масштабируемую обслуживаемую очередь для хранения сообщений при передаче их между компьютерами.

» **FPS (Flexible Payment Service)** Сервис позволяет коммерческим сайтам принимать платежи через существующие учетные записи Amazon.

Завести учетную запись на AWS нетрудно, особенно если у вас уже есть учетная запись на Amazon, потому что вы сможете платить за web-сервисы той же картой, с которой платите за книги и другие товары. Однако в процессе регистрации есть несколько необычный этап, когда вам звонит робот-автомат и просит набрать пин-код с экрана компьютера на клавишах телефона.

> «Оплачивается время от создания ВМ до ее удаления.»

› **На первом шаге мастера создания виртуальных машин можно выбрать систему из образов, предложенных сообществом, или создать собственный.**



› **В финальном окне мастера можно выбрать тип экземпляра и указать их количество.**

Покрыто облаками

Сразу после входа в систему мне не терпелось создать несколько виртуальных машин. Сначала я сгенерировал открытый и закрытый ключи для аутентификации по SSH на моих ВМ. Можно создать несколько пар или пользоваться одной и той же на всех машинах. Открытый ключ «встраивается» в создаваемые вами экземпляры ВМ, позволяя выполнять безопасный вход в систему по SSH с помощью закрытого ключа, который хранится в файле на локальном компьютере. Как и следовало ожидать, существует web-мастер для создания новых виртуальных машин, и состоит он из двух основных шагов: выбора системы и оборудования.

Чтобы упростить первый этап, Amazon предоставляет несколько образов ВМ для быстрого старта: базовая версия Fedora, Fedora с LAMP, различные конфигурации Windows 2003 и т.д. В качестве альтернативы можно выбрать один из более чем трех тысяч образов, предоставленных сообществом – множество систем на основе Fedora, Ubuntu, Debian, CentOS, SUSE, Solaris и Windows (зайдите на http://aws.amazon.com и кликните на Amazon Machine Images). Они могут стать отправной точкой для вашей собственной конфигурации – так как у вас есть права root, вы можете установить любое другое ПО, как и на обычном компьютере. Можно даже создавать и сохранять собственные образы Amazon Machine и запускать с их помощью дополнительные экземпляры ВМ.

Конфигурация оборудования выбирается из нескольких вариантов. Начальный уровень, Small, представляет собой 1,7 ГБ оперативной памяти, 160 ГБ на жестком диске и один вычислитель-

Как насчет поддержки?

Базовая поддержка, которая предоставляется бесплатно после входа в систему, состоит из дюжины форумов, которые, похоже, отличаются активностью и хорошим соотношением «сигнал/шум». Есть также несколько FAQ. В качестве альтернативы можно заплатить за поддержку Silver или Gold, 100 или 400 долларов в месяц соответственно. Они предоставляют канал связи «один-на-один» и неограниченное число обращений.

Обзорный тур

Прежде чем входить в систему, осмотритесь. Даже не имея учетной записи, можно получить хорошее представление о том, как выглядит консоль управления AWS, зайдя на http://aws.amazon.com/console и кликнув «Take The Tour».

ный модуль с 32-битным процессором. В вариантах Large и Extra Large объем ОЗУ, места на диске и число процессоров постепенно увеличиваются. Для этих вариантов есть версии с большим числом процессоров, предоставляющие до 20 вычислительных блоков (мощность одного вычислительного блока эквивалентна процессору 2007 Opteron или 2007 Xeon частотой 1–1,2 ГГц). В последнем окне мастера есть поле, в которое нужно ввести количество экземпляров – создать 99 не сложнее, чем один!

Консоль управления

Виртуальные машины создаются за сущие минуты. После этого они управляются с административной консоли AWM – web-интерфейса, где можно просматривать состояние ВМ, перезагружать, клонировать и останавливать (уничтожать) их. Можно создавать (виртуальные) диски в Elastic Block Store и подключать их к ВМ.

Все, что делается в консоли управления, допускается делать и с командной строки – с помощью набора утилит на Java, загружаемых в виде zip-файла. Так, команды

```
$ ec2-run -instances ami-235fba4a -k gsg-keypair
RESERVATION   r-7430c31d   924417782495   default
INSTANCE   i-ae0bf0c7   ami-2bb65342   pending gsg-
keypair   0   m1.small 2008-03-21T16:19:25+0000   us-east-1a
```

запустят виртуальную машину **ami-235fba4a**. Команда

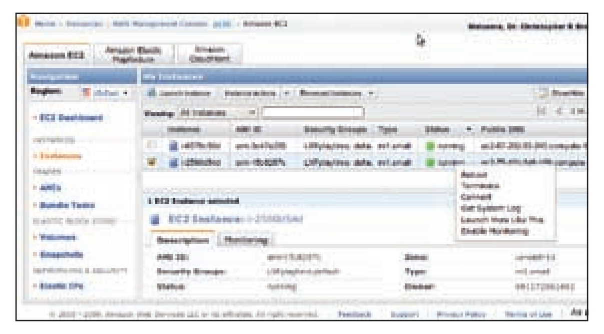
```
$ ec2-terminate-instances i-ae0bf0c7
```

остановит виртуальную машину.

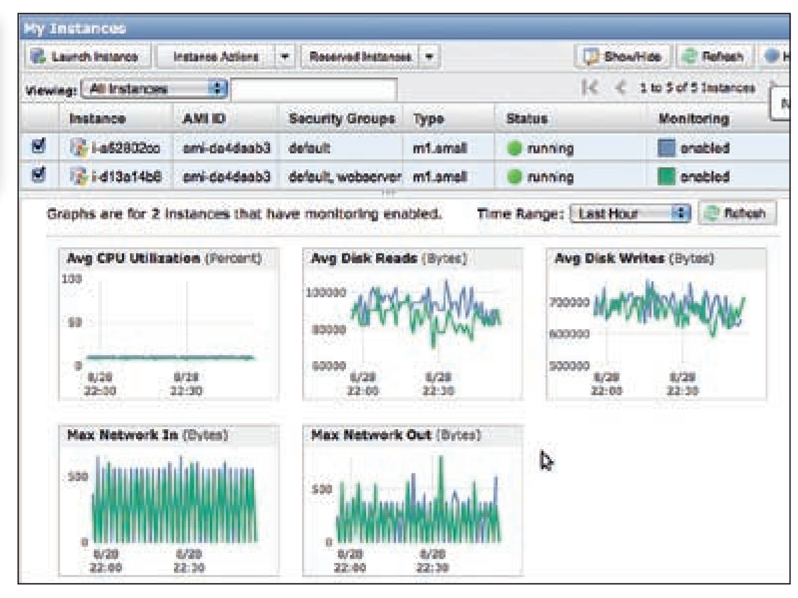
Диск EBS (см. ниже) создается аналогично – командой **ec2-create-volume**, а список всех образов выводится командой **ec2-describe-images**. Также есть модули расширения для IDE *Eclipse*, которые позволяют, например, запустить кластер серверов *Tomcat* и одним щелчком загрузить туда ваши приложения.

У каждого запускаемого вами экземпляра ВМ будет публичный IP-адрес и публичная запись DNS, но вид у нее малосимпатичный: нечто вроде ec2-79-125-4-198.eu-west1.compute.amazonaws.com. Хотя можно создать в DNS псевдоним (запись CNAME) и связать с этим адресом ваше доменное имя. Альтернатива – арендовать эластичные IP-адреса (Elastic IPs, см. выше). Это постоянные публично доступные адреса; их можно подключить к любому экземпляру ВМ, а при желании перераспределить их среди своих ВМ – в консоли управления это сделать нетрудно.

Выключив ВМ, избежать почасовой оплаты не удастся (хотя ВМ можно перезагрузить). Оплачивается время от создания ВМ до ее удаления – виртуального эквивалента выбрасывания компьютера в окно. При этом подключенное к машине локальное хранилище удаляется. Однако можно создать диски EBS размером от 1 ГБ до 1 ТБ. Эти диски похожи на логические тома: ядро видит их как блочные устройства типа **/dev/sdf**; разместить файловую систему можно и на них. Можно даже создать несколько EBS-дисков, собрав из них RAID-массив. Диски EBS не зависят ни от одной из ВМ, и их можно подключить к любому экземпляру (хотя только к одному в каждый момент времени). Можно также делать снимки состояния дисков EBS, делиться ими с другими участниками сообщества и создавать из них новые диски. Для EBS автоматически выполняется резервное копирование (внутри облака), чтобы избежать проблем с «единственной точкой отказа».



› **Из консоли управления AWS легко создавать, просматривать и удалять виртуальные машины, управлять дисками EBS и делать многое другое.**



› **Cloudwatch позволяет построить график использования ресурсов компьютера за период времени до двух недель.**

Запустив мониторинг для виртуальных машин, вы получите графики использования ресурсов – процессора, чтения и записи на диск, входящий и исходящий сетевой трафик. Графики могут охватывать период времени от часа до двух недель.

Я не претендую на нечто большее, чем весьма поверхностный рассказ об облаке EC2. Я счел сайт Amazon весьма приятным и удобным, и с удовольствием обнаружил, что это также относится к консоли управления. И я изрядно повеселился – и все по цене кружки пива. LXF

Прайс-лист

А сколько же все это стоит? На этот вопрос нет простого ответа. Здесь нет минимальной платы и долговременных контрактов – вы платите за каждый час работы ВМ. Вариант начального уровня Small стоит 11 центов в час, самый продвинутый вариант с большим числом процессоров Extra Large – 88 центов в час. (Не ищите калькулятор: это 80 и 640 долларов в месяц соответственно.) Можно также арендовать зарезервированные ВМ, за которые платится ежегодный взнос и (гораздо меньшая) почасовая оплата. Например, приобретенный по запросу вариант Small обходится в 964 доллара в год, а в виде зарезервированной виртуальной машины – 577 долларов.

Elastic Block Store стоит 10 центов за ГБ в месяц плюс плата за обмен данными – 10 центов за миллион запросов ввода/вывода. (А вы знаете, сколько запросов ввода/вывода вы делаете? Я – нет.)

Также нужно платить за сервисы типа Amazon S3, мониторинг Cloudwatch и эластичные IP-адреса, и все это способно вылиться в немалую сумму. По ссылке **http://calculator.s3.amazonaws.com/calc5.html** вы найдете месячный калькулятор, который посчитает результат, но вам все равно придется вводить ожидаемые значения для жестких дисков и сетевого трафика, оценить которые может быть непросто.

Сравним эти цены с арендой выделенного сервера: примем, что ее можно приобрести долларов за 100 в месяц. Вы получаете компьютер, который вроде бы не сильно отличается от варианта Small. Но это «вроде бы» заслоняет от нас суть. «Эластичность» Amazon EC2 позволяет быстро нарастить (и уменьшить масштабы) вычислительных ресурсов в соответствии с вашими потребностями. Например, можно увеличить емкость своих вычислительных ресурсов на месяц, чтобы сайт отвечал требованиям, вызванным проведением рекламной кампании, и ликвидировать все по ее окончании.

LINUX FORMAT Учебники

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Начинал с Агатов. Когда-то даже знал, что такое Робик.

Сообщество по интересам

Я использую GNU/Linux на своем компьютере исключительно по той причине, что мне так удобно. Но удобство программного обеспечения является не единственным плюсом такого выбора. Обычно в голову приходит еще и бесплатность свободного ПО. Это действительно полезно, но сейчас я хотел бы акцентировать внимание на сообществе, которое окружает феномен GNU/Linux.

Причем я не стал бы зацикливаться именно на GNU/Linux – это просто узнаваемая вывеска, под которой легко объединяется множество других сообществ открытого ПО. Обмен знаниями вынуждает идти на некоторые жертвы и усилия, но только так можно доказать в этом сообществе свою значимость.

Это полезная для прогресса способность. Да и просто люди приятные попадаются: нет-нет, да и ловишь себя на том, что «массу гравитационного поля» гораздо интереснее обсуждать на linux.org.ru, чем на специализированном форуме.

P.S. Этим летом игру *Seven Kingdoms* (http://7kfans.com/) – стратегию реального времени с дипломатией – выпустили под GPL. Игра не из новинок, но некоторым нравится.

P.P.S. Компания «Аскон» осторожно интересуется возможностью запуска *Компас 3D* в среде GNU/Linux.

E.m.Baldin@inp.nsk.su

В этом месяце вы научитесь...

Совет месяца: Смерть пожирателям трафика!



На вашем рабочем столе ведь есть системный монитор сетевого трафика – что-то типа *Gkrellm*? Если да, то вы наверняка хоть раз глядели на него, недоумевая, откуда взялись все эти мегабайты. Есть несколько программ, работающих как *top*, но в отношении сети; часть из них требует времени на сбор статистики и предоставлените отчета по протоколам и портам. А если вам просто нужно узнать, кто сейчас захватил весь канал, попробуйте *Nethogs* (nethogs.sf.net).

Это простое приложение, выводящее объемы принятых и переданных за секунду данных, разбитые по процессам. *Nethogs* обычно устанавливается в **/usr/sbin**, и его необходимо запускать от имени root, напрямую или через **sudo**. По умолчанию программа наблюдает за интерфейсом eth0; если вас интересует какой-либо другой, просто укажите его в качестве аргумента. Можно вести наблюдение за несколькими устройствами одновременно.

Nethogs выводит для каждого процесса потребление трафика, идентификатор (чтобы вы могли послать главному обжоре **SIGTERM**) и интерфейс, который он использует. Для управления программой нужны всего две кнопки: m переключает между отображением скорости загрузки/выгрузки и всего трафика, потребленного за время работы программы; Q завершает работу *Nethogs*.

Chromium: Идем

Chromium, открытая версия *Chrome* от Google – браузер, сияющий на фоне других. **Энди Ченнел** покажет, на что он способен…





Наш эксперт

Энди Ченнел
Энди, видимо, будет делать Первые шаги в Linux вечно, а технологии он заинтересовался, открыв для себя Dragon 32.

Выход Google на рынок браузеров был в общем-то неизбежен, но все-таки стал сюрпризом. Пока *Chrome* никого серьезно не потеснил, но все может измениться уже в этом году, после выпуска Chrome OS и другой новинки – Google Wave. Версии Chrome для Linux еще нет, и мы пройдемся по установке *Chromium* – это сборка со свободным кодом от разработчиков – в системах на базе RPM и Deb, а также рассмотрим некоторые его возможности.

Двоичные файлы для Fedora 10 и 11 доступны на http://spot.fedorapeople.org/chromium. После установки, выполняемой двойным щелчком, браузер появится в вашем меню Интернет [Internet]. Чтобы всегда иметь последнюю версию, вы можете добавить репозиторий *Chromium* в *Yum*. Для этого откройте текстовый редактор и наберите следующий код:

```
[chromium]
name=Chromium Test Packages
baseurl=http://spot.fedorapeople.org/chromium/F$releasever/
enabled=1
gpgcheck=0
```

Сохраните его на рабочем столе под именем **chromium.repo**. Затем откройте *Nautilus* от имени root. Откройте терминал и введите

```
su nautilus
```

У вас спросят пароль root, а затем запустится *Nautilus*. Учтите, что менеджер файлов запустится с привилегиями root, и вы можете ненароком удалить или повредить некоторые жизненно важные системные файлы. Будьте осторожны!



› Сверхбыстрый движок *Chrome* на Javascript был разработан с прицелом на службы типа Wave.

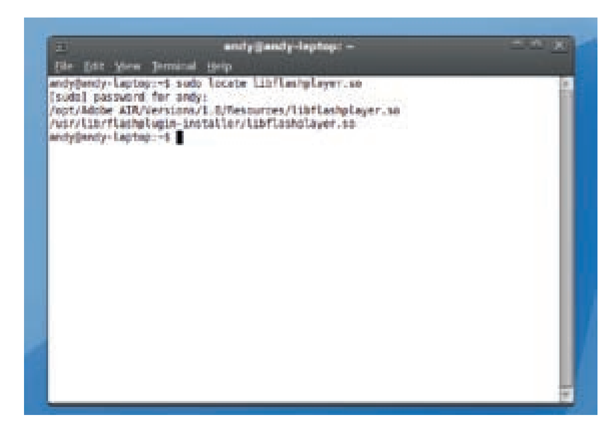
Скопируйте **chromium.repo** в **/etc/yum.repos.d/**. По завершении закройте *Nautilus* и вновь откройте терминал. Теперь можно установить *Chromium*, введя

```
yum install chromium
```

Этот способ установки дольше, но зато при каждом обновлении репозитория вы будете мгновенно получать доступ к новой версии.

Установка в Ubuntu

Ubuntu – один из дистрибутивов, который, похоже, будет «официально» поддерживаться *Chrome* после финальной окончательной версии. А пока приложение имеет собственную PPA-страницу, поэтому его очень просто установить. Сперва откройте Synaptic (Система > Администрирование > Менеджер пакетов Synaptic [System > Administration > Synaptic Package Manager]), перейдите в Настройки > Репозитории [Settings > Repositories] и затем



› До установки модуля расширения Flash следует проверить: вдруг он уже присутствует в системе.

» **Месяц назад** Мы порулили привилегиями пользователей и обновили ПО.

в Сеть

на вкладку Сторонние приложения [Third Party Software]. Нажмите Добавить [Add] и скопируйте в строку адреса следующее:

```
deb http://ppa.launchpad.net/chromium-daily/ppa/ubuntu jaunty main
```

Неплохо также добавить репозиторий с исходными текстами:

```
deb-src http://ppa.launchpad.net/chromium-daily/ppa/ubuntu jaunty main
```

Отметим, что предпоследняя часть этих строк должна содержать кодовое имя вашей версии Ubuntu (**hardy**, **intrepid**, **jaunty** или **karmic**).

Нажмите кнопку Закрыть [Close], а затем Обновить [Reload]. В завершение найдите **chromium-browser** при помощи строки поиска и установите последнюю версию. Приложение появится в меню Интернет. *Chromium* лучше всего работает с установленными шрифтами Microsoft Core Fonts, и если их у вас пока нет, поищите **msttcorefonts** и установите соответствующий пакет до использования *Chromium*.

Добавим поддержку Flash

По умолчанию, *Chromium* не может проигрывать Flash-содержимое, что обидно. К счастью, можно включить его поддержку, ненадолго переместившись в консоль. Проверьте, не установлен ли уже Flash, и найдите его местоположение в файловой системе. Откройте терминал и введите **sudo locate libflashplayer.so**. Если Flash установлен, то вы увидите нечто вроде

```
/opt/Adobe AIR/Versions/1.0/Resources/libflashplayer.so
/usr/lib/flashplugin-installer/libflashplayer.so
```

Чтобы заставить Flash работать, скопируйте **libflashplayer.so** из одного из этих мест в каталог **plugins** внутри рабочей директории *Chromium*. Путь к ней – **/usr/lib/chromium-browser/plugins/**.

Скопируем файл из первого каталога в указанный:

```
sudo cp /usr/lib/flashplugin-installer/libflashplayer.so /usr/lib/chromium-browser/plugins
```

Теперь можно запустить *Chromiun* с включенным Flash, введя **chromium-browser --enable-plugins**, но эффективнее будет добавить для выполнения этого действия команду меню. Сделайте это в стандартном редакторе меню вашего дистрибутива, который, вероятно, располагается в Настройках [Preferences]. Найдите



› Вкладки-иконки снижают захламленность вашей панели, чтобы вы могли захламлять ее дальше.



› Изменяйте расположение миниатюр на стартовой странице, перенося их на новые места.

пункт меню Chromium-Browser, щелкните по нему правой кнопкой мыши для редактирования Свойств [Properties] и введите **--enable-plugins** для команды, запускающей данное приложение. Сохраните и закройте диалог, а затем выберите в меню пункт *Chromium*.

Советы по использованию

Установив *Chrome* и модуль расширения Flash, мы можем рассмотреть некоторые классные функции, ставшие нам доступными.

Если вам не нравится стандартный вид *Chromium* от Google, щелкните на изображении гаечного ключа в верхнем правом углу окна браузера и выберите Опции [Options], а затем вкладку Личные предпочтения [Personal Stuff]. В разделе Темы [Themes] нажмите Получить темы [Get Themes]. Появится новая вкладка с набором тем; выберите одну из них, нажмите Применить тему [Apply Theme] и посмотрите, как изменился внешний вид браузера.

При запуске *Chromium* появляется Google-эквивалент экспресс-панели Opera. В верхнем правом углу вы увидите опции для изменения вида с Миниатюры [Thumbnail] на Список [List]. В режиме миниатюр отображаются экранные снимки последних посещенных страниц. Их можно перемещать, просто перенося мышью; удалять миниатюры, поднося к ним курсор мыши и затем нажимая на иконку в верхнем правом углу; или закреплять на странице в данной позиции, при помощи булавки там же. Под миниатюрами представлен список недавно закрытых вкладок и окон, которые можно вновь открыть одним щелчком.

Вкладки *Chromium* великолепны. Вы можете перетаскивать их по панели в другое место, «вытягивать» на рабочий стол, чтобы создать новое окно, или перемещать существующие на панель вкладок. Если у вас обычно открыто множество вкладок, это может привести к захламлению. Уменьшите постоянно открытые вкладки, щелкнув по ним правой кнопкой мыши и выбрав в меню пункт Вкладка-иконка [Pin Tab]. Это приведет к смещению вкладки к левому краю и сворачиванию в пиктограммы.

Для просмотра в приватном режиме, щелкните на иконке гаечного ключа и выберите Создать новое защищенное окно [Create New Incognito Window]. Откроется новое окно (слегка иного вида и с аватарой шпиона), позволяющее просматривать сайты без записи истории посещений – история, cookie и кэш основного браузера останутся нетронутыми. Однако загрузки и новые закладки сохранятся.



Если вы установили официальную Google-сборку *Chrome* для Linux, модуль расширения Flash надо поместить в **/opt/Google/chrome/plugins**. Возможно, этот каталог придется предварительно создать.

OpenSUSE: Мы

Обновление системы – важная задача в обеспечении безопасности любого дистрибутива.

Скорая помощь

Обновления содержат не только новые версии приложений. Они также могут исправить важные проблемы безопасности – пожалуйста, не отключайте их.

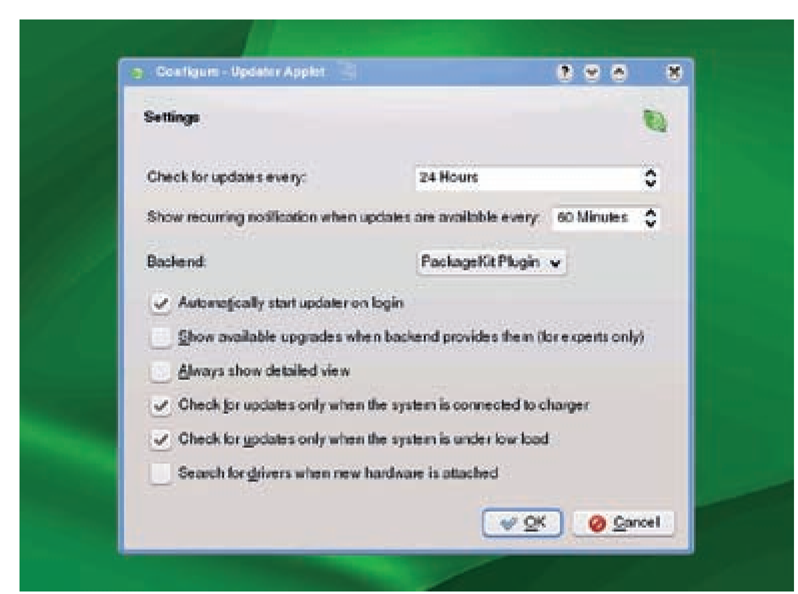
OpenSUSE – это общедоступная версия коммерческого дистрибутива Linux от Novell. Она создается сообществом и инженерами Novell и дает пользователям хорошее представление о следующей платной версии продукта. На данном уроке мы пробежимся по различным способам поиска и установки новых приложений и поддержания вашей системы OpenSUSE в актуальном состоянии.

Текущей является версия OpenSUSE 11.2. Она поставляется в нескольких вариантах; главные из них – Live CD и издание на DVD, с большим количеством приложений. В этом учебнике мы будем использовать версию Live CD, установленную на машину. Это означает, что главным источником новых приложений для нас будут сетевые репозитории OpenSUSE, и жизненно важно наличие широкополосного соединения. Владельцы DVD могут следовать той же процедуре; разве что в процессе установки, вероятно, будут появляться запросы вставить соответствующий DVD.

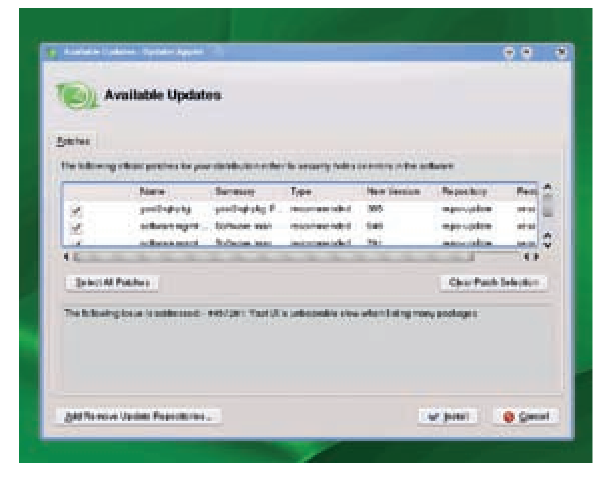
Однако начнем мы с обновлений. По умолчанию OpenSUSE настроена на автоматическое слежение за ними, и если что-то появится, в области уведомлений нижней панели KDE вы увидите небольшую оранжевую звезду. Для запуска процесса щелкните по ней. Диалог обновления достаточно скуден, но если вы нажмете кнопку Детали [Details], то увидите дополнительную информацию о том, что будет загружено и установлено.

Вы в ответе

Эта опция позволяет также отменить обновление, если, например, оно слишком велико, а у вас медленное соединение, или если вы просто не желаете обновлять конкретное приложение. Просто снимите галочку около элемента, который не желаете устанавливать, и нажмите кнопку Установить [Install]. Перед началом загрузки будет выдан запрос пароля root.



›Утилита обновления SUSE имеет несколько параметров настройки, удобных, если у вас не широкополосное соединение.



›Используйте кнопку Детали [Details] для получения подробной информации о доступных обновлениях, включая время на их загрузку.

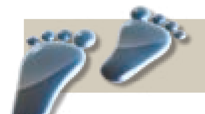
По умолчанию, SUSE просматривает обновления каждые 24 часа, а система обновления запускается при каждом вашем входе. Эти настройки можно изменить, щелкнув правой кнопкой мыши на зеленом гекконе в системном лотке и выбрав Настроить апплет [Configure Applet]. В верхнем разделе определяется период обновлений и время между подачей уведомлений – полезно, если обновления не всегда следует выполнять с ходу, но вы все же желаете получать напоминания о доступности новых версий.

Больше выбора

Вы можете также решить, следует ли выводить детализированный вид автоматически – нам так нравится больше – а также настроить условия, при которых обновление начнет свою работу. Для ноутбука удобно выбрать пункт When The System Is Under A Low Load [При низкой загрузке системы]. Последняя опция, автоматически загружающая и устанавливающая драйвера при подключении нового устройства, превосходна для тех, кто ежедневно пробует множество периферии. По завершении нажмите OK, и система обновления продолжит свою работу безмолвно, приставая к вам только в случае появления новых продуктов.

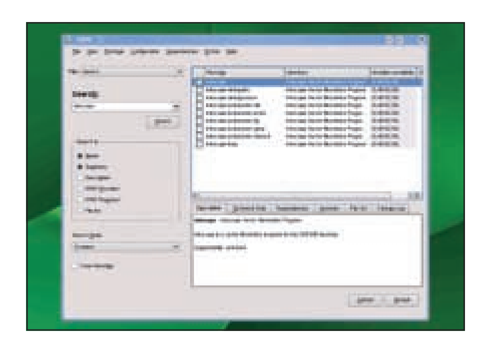
Если вам неохота ждать запуска менеджера обновлений системой, инициируйте проверку самостоятельно, щелкнув правой кнопкой мыши на иконке обновлений и выбрав Check Now [Проверить немедленно]. Утилита сравнит ваши приложения с последними изменениями репозиториев SUSE и сообщит о том, что следует сделать. Если вы вообще не желаете получать уведомления об обновлениях, щелкните на иконке правой кнопкой мыши и выберите Quit [Выход]. Это приведет к остановке менеджера обновлений для текущей сессии, но при следующем входе он будет запущен вновь – если только вы не отмените его автоматический старт в диалоге настройки. Если позднее вы пожелаете перезапустить утилиту обновлений, ищите ее в пункте Software [Приложения] меню Administration Settings [Администрирование].

освежаемся



Шаг за шагом: Добавляем ПО к OpenSUSE

В OpenSUSE есть тысячи программных пакетов, от игр до генеалогии. Главное – знать, как их заполучить. Данное руководство расскажет об этом.





1 Используем SUSE

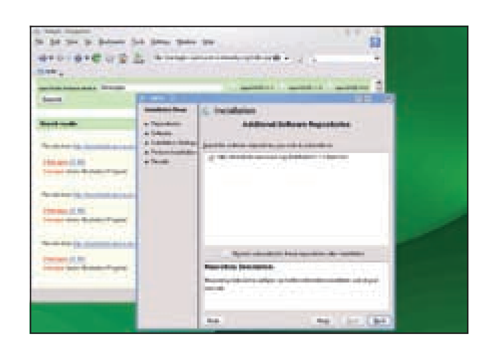
Первое, к чему следует обратиться – собственный менеджер пакетов SUSE. В KDE откройте меню панели и посмотрите в разделе Computer [Компьютер]. Вам необходим пункт Install Software [Установить приложения]. После ввода пароля root откроется менеджер пакетов. Используйте строку Search [Поиск] для поиска приложения, а затем выберите его в списке слева. В нижней части окна появится описание программы.

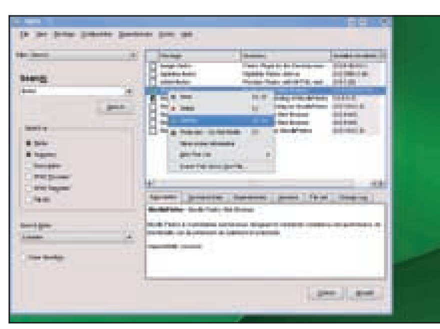
2 Добавляем репозиторий

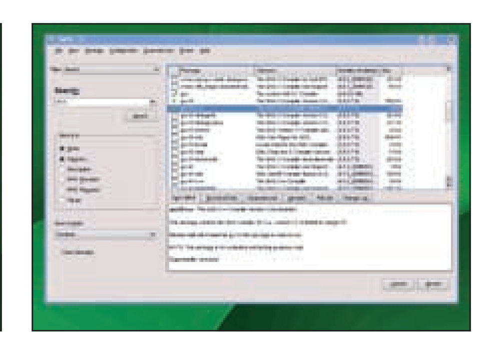
Как Ubuntu и Fedora, SUSE может использовать новые репозитории для установки и обновления программ. В менеджере пакетов выберите Configuration > Repositories [Настройка > Репозитории], а затем нажмите кнопку Add [Добавить], расположенную у нижнего края окна. SUSE умеет работать с различными дистрибутивными носителями. Выберите его тип (в нашем случае это URL), затем нажмите кнопку Next [Далее] для ввода соответствующих деталей – web-адреса, пути к CD и т.д. Крутые линуксоиды могут создавать локальные репозитории.

3 Ищем RPM

OpenSUSE использует пакеты формата Red Hat Package Manager, или RPM. Они также применяются в Red Hat, Mandriva Linux и других, поэтому необходимо загрузить подходящую версию для вашего релиза SUSE. Среди популярных источников RPM – RPMfind.net, Sourceforge и сайты приложений. После загрузки RPM вы можете дважды щелкнуть на нем для запуска установщика. Однако это не лучший способ инсталляции: он приводит к проблемам с зависимостями.







4 Установка одним щелчком

SUSE имеет собственную систему установки одним щелчком, и в ней доступно множество приложений. Для поиска пакетов перейдите на http://packages.opensuse-community.org и ищите по названию пакета и версии дистрибутива. Щелчок на иконке установки приведет к загрузке и добавлению пакета в систему. Мы сделали это в *Konqueror*, но не в *Firefox*, в котором были некорректные настройки. Этот метод должен разрешать проблемы зависимостей.

5 Ручные операции

Вы можете удалить или вручную обновить пакеты, перейдя в главный интерфейс (Panel > Computer > Install Software) [Панель > Компьютер > Установить приложения] и запустив поиск того, что следует изменить, а затем щелкнув правой кнопкой мыши на галочке в колонке слева от имени пакета. Вы увидите простое меню. Большее число настроек доступно после выбора пункта All In This List [Для всех элементов списка].

6 Если ничего не получается...

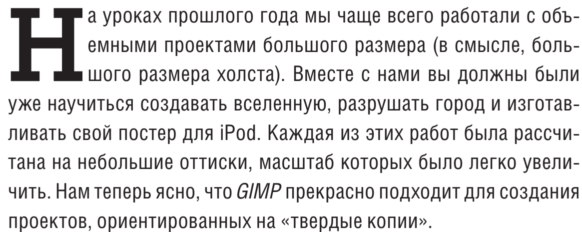
Если двоичной сборки желаемого пакета нет, можно собрать приложение из исходных текстов. Сначала установите компилятор C/C++ и инструменты через менеджер пакетов. Затем загрузите и распакуйте файл с исходными текстами в терминале при помощи команды **tar -zxvf filename.tgz**, выполните **./configure** для настройки исходников, затем **make** и, наконец, **make-install**. Последнюю команду следует дать от имени root. LXF

» **Через месяц** Обмениваемся мыслями в Twitter и резервируем файлы онлайн.

GIMP: Иконки

Рабочий стол в духе Mac OS X начинается с профессионально сделанных иконок. **Майкл Дж. Хэммел** создаст в *GIMP* подобный дизайн одной левой.





Наш эксперт

Майкл Дж. Хэммел участвует в разработке *GIMP* и является автором трех книг по данной тематике, включая самую новую – «Эффекты в *GIMP*: руководство для художников».

На уроках прошлого года мы чаще всего работали с объемными проектами большого размера (в смысле, большого размера холста). Вместе с нами вы должны были уже научиться создавать вселенную, разрушать город и изготавливать свой постер для iPod. Каждая из этих работ была рассчитана на небольшие оттиски, масштаб которых было легко увеличить. Нам теперь ясно, что *GIMP* прекрасно подходит для создания проектов, ориентированных на «твердые копии».

Но печатные проекты – только частица того, на что способен *GIMP*. Этот многоцелевой инструментарий подходит для различных дизайнерских начинаний – например, работы с цифровыми фото. Я не часто касался фотографии просто потому, что мало фотографирую – у меня даже нет цифровой камеры; но это мой личный просчет. *GIMP* активно используется в областях обработки изображений и поддерживает многочисленные форматы RAW (через модуль расширения *ufraw*). Это очень удобно для фотографов, поскольку качество RAW-изображений несравнимо лучше качества снимков в формате JPEG, которые, как правило, выдает цифровая камера. Следовательно, возможность работать с RAW означает конечный продукт более высокого качества.

Первоисточники

Данный урок базируется на превосходном пособии по *Photoshop* Ричарда Карпентера [Richard Carpenter] из HV-Designs, под названием ‘Carbon Style Icon’.

» http://hv-designs.co.uk/2009/02/10/carbon-style-icon
» *SimDock*: http://simdock.sourceforge.net
» *Avant Window Navigator*: https://launchpad.net/awn



› **Используйте инструменты Масштаб и Высота дорожки для улучшения предпросмотра содержимого вашего видеоряда.**

На другом полюсе изобразительного ряда располагается настольная графика – в частности, иконки. Иконка – это маленькое изображение, используемое для обозначения программы, файла, устройства или данных, доступных для пользователя. Чтобы иконки не заполонили рабочий стол, программы типа *Window Navigator* и *SimDock* помогают организовать их в доки а-ля Mac. Эти доки похожи на панели KDE или Gnome, но только в трехмерном формате. *SimDock*, в частности, вообще разработан так, чтобы максимально походить на док в Mac OS X.

Доки – это, по сути, менеджеры иконок, автоматически управляющие изменением размера последних. *SimDock*, например, по умолчанию отображает иконки в очень мелком масштабе и увеличивает их по наведению курсора мыши. Пользователи могут указать степень увеличения в настройках, и при большем коэффициенте иконки примут более крупный размер.

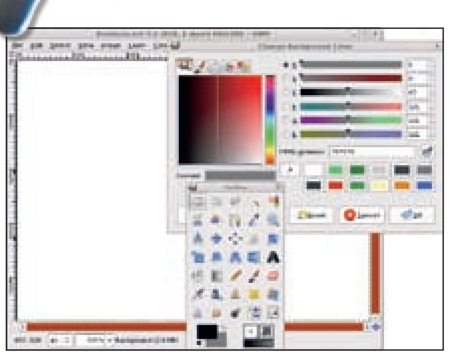
Вот мы и добрались до *GIMP*. Для больших иконок проще сделать рисунок-основу, поскольку *GIMP* умеет работать с холстами большего размера. Это легче и сильно отличается от работы, скажем, с иконками панели запуска в GNOME, где все они имеют одинаковый размер – как правило, не превышающий 32×32 пикселя. *SimDock* позволяет создать иконки гораздо большего размера и берет на себя автоматическое масштабирование.

На сей раз мы отправляемся в мир иконок, и я проведу вас через процесс создания пиктограмм, пригодных для *SimDock* и других доков. Рецепт рассчитан на *GIMP 2.6* и годится даже для начинающих *GIMP*-художников; достаточно только базовых навыков и понимания раскладки панели инструментов. Сторонние стоковые изображения нам не понадобятся. Продвинутым пользователям потребуется на работу менее часа; у новичков времени уйдет чуть больше.

» **Месяц назад** Мы вместе создали дизайн по мотивам Франка Миллера.

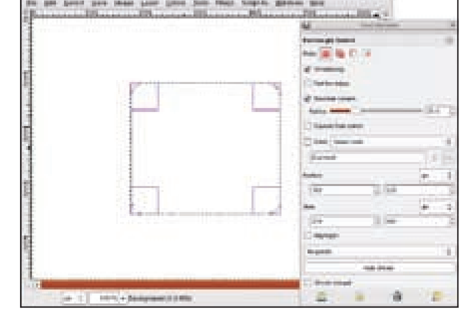
в стиле Leopard

Шаг за шагом: Создаем иконку



1 Зададим цвета

Откроем новое изображение размером 640×480 пикселей (Файл > Создать). Восстановим значения цветов фона и переднего плана, нажав клавишу D в пределах окна изображения. Щелкнем по квадратику цвета фона на панели инструментов и в диалоге настройки цвета фона выставим значения 121, 121 и 121 для R, G и B соответственно. Закроем диалог выбора цвета. Серый и черный цвета теперь будут использоваться в качестве градиента фона для законченной иконки.



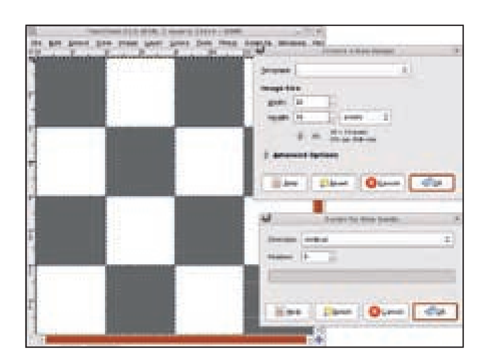
2 Выделим прямоугольник

Добавим новый прозрачный слой (Слой > Создать слой) и назовем его «фон». На панели инструментов выберем прямоугольное выделение и в диалоге параметров инструмента отметим Закругленные углы с радиусом 25. Сделаем начальное выделение в окне изображения и в окне параметров инструмента выставим размеры 274×240 и позицию 183×120. Это размер нашей иконки плюс некоторые прозрачные поля.



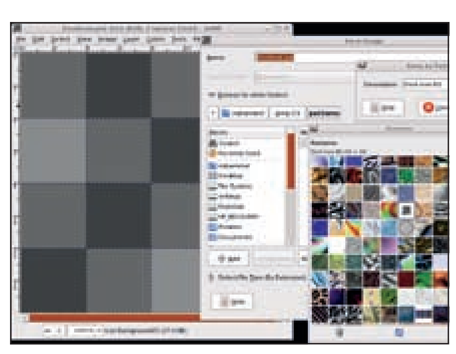
3 Градиентная заливка

Создадим посредине изображения направляющую. Выберем инструмент Заливка и выставим режим Нормальный, непрозрачность 100 %, градиент Основной в фоновый со снятой галочкой Развернуть и линейной формой. Протащим сверху вниз, вдоль направляющей. Теперь уберем выделение (Выделение > Снять) и добавим «отбрасываемую тень» (Фильтры > Свет и тень > Отбрасываемая тень), со смещением в 3 пикселя и размыванием 5.



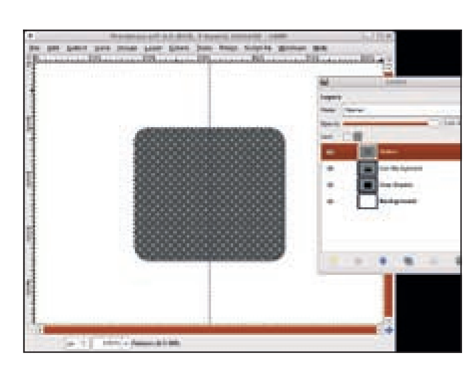
4 Шахматная клетка

Откроем новое окно изображения 16×16 пикселей и 10 раз нажмем клавишу «+», увеличивая масштаб. Добавим вертикальные направляющие (Изображение > Направляющие > Новая направляющая) с положениями 4, 8 и 12 и горизонтальные направляющие с положениями 4, 8 и 12 пикселей. Выставим цвет переднего плана для RGB на 103, 103, и 103. При помощи прямоугольного выделения прорисуем блоки и по очереди зальем их цветом переднего плана.



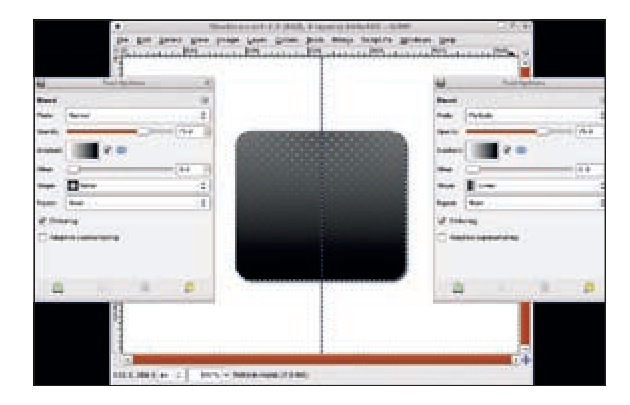
5 Завершаем клетки

Изменим значения RGB цвета переднего плана на 67, 67 и 67, затем выделим и зальем квадраты 1/2, 1/4, 2/3, 3/2, 3/4 и 4/1. Изменим значения RGB цвета переднего плана на 141, 141 и 141 и зальем квадраты 2/1 и 4/3. Сохраним этот файл под названием **DockIcon.pat** в каталоге **$HOME/.gimp-2.6/patterns**, затем откроем диалог шаблонов (Окна > Панели > Шаблоны) и щелкнем по иконке Обновить, чтобы добавить новый шаблон в список. После этого закроем окно **DockIcon.pat**.



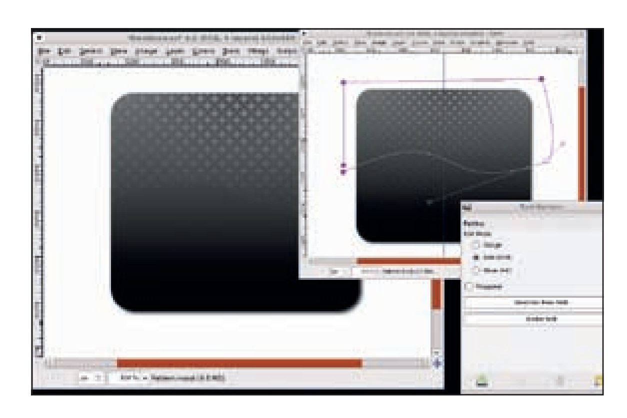
6 Добавим шаблон

Добавим прозрачный слой к изображению иконки и назовем его Шаблон. Щелкнем по слою Фон в диалоге слоев и создадим выделение (Слой > Прозрачность > Альфа-канал в выделение). Щелкнем на слое Шаблон, чтобы он стал активным. В диалоге Шаблоны (Окна > Панели > Шаблоны) перетащим наш шаблон Dock Icon BG в окно изображения, чтобы заполнить выделение. У нашей иконки теперь есть клетчатый фон. »



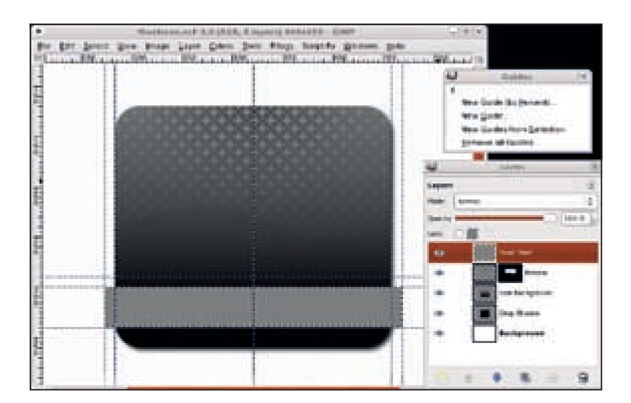
7 Маскируем шаблон

Добавим белую маску слоя к слою Шаблон. Нажмем D в пределах окна изображения, выберем инструмент Заливка, настроим непрозрачность на 75 % и радиальную форму. Протащим от верхнего края иконки до середины. Переключим форму на линейную и режим на Умножение, затем протащим от середины иконки до низу и снова сверху до середины.



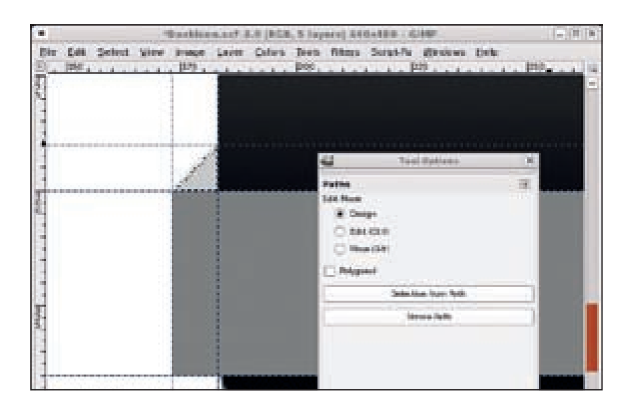
8 Другая маска

Применим маску слоя, уберем выделение, затем добавим еще одну белую маску к слою Шаблон. С помощью инструмента Кривые разместим контрольные точки так, чтобы получился прямоугольник, очерчивающий верхнюю половину иконки, затем передвинем рукоятки контрольных точек, чтобы получить волнистый эффект вдоль нижней границы кривой. Далее инвертируем выделение и зальем его черным.



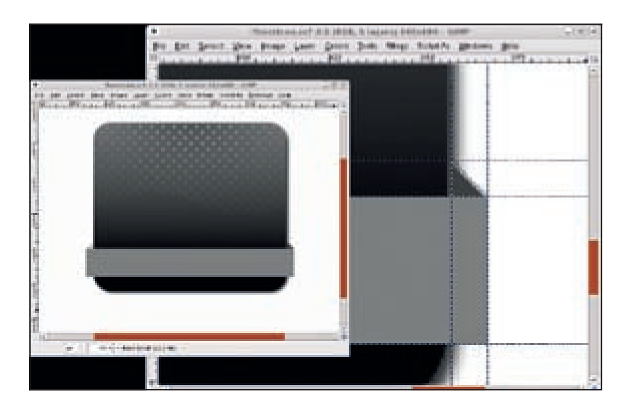
9 Передний выступ

Добавим прозрачный слой на самый верх стопки слоев и назовем его «Передний Выступ». Создадим вертикальные направляющие с положениями 173, 183, 458 и 468 и горизонтальные – на 290, 300 и 340, затем сделаем прямоугольное выделение от точки пересечения направляющих 173/300 до 468/340. Настроим цвет переднего плана на нейтральный серый.



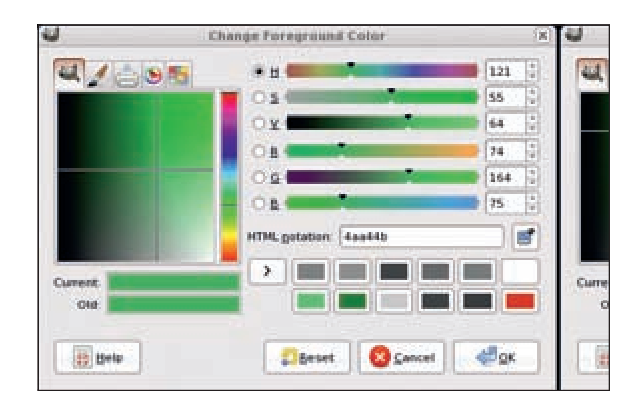
10 Левый выступ

Увеличим масштаб и наведем фокус на левую сторону выступа. Выберем Контуры и тип действия – Создание. Отметим направляющие на точках пересечения 173/300, 183/290 и 183/300 и щелкнем по кнопке Выделение из контура в параметрах инструмента. Настроим цвет переднего плана на более светлый серый, со значением 203 для R, G и B. Перетащим цвет переднего плана в выделение.



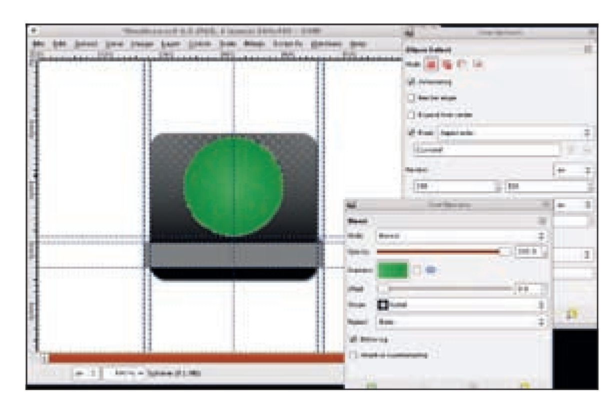
11 Правый выступ

Переместимся на правую сторону изображения, чтобы увидеть правый край переднего выступа. Выберем Контуры и щелкнем по пересечению направляющих 458/290, 468/300 и 458/300. Нажмем на кнопку Выделение из контура в диалоге параметров инструмента. Настроим цвет переднего плана на темно-серый, перетащим его в выделение, затем уберем выделение.



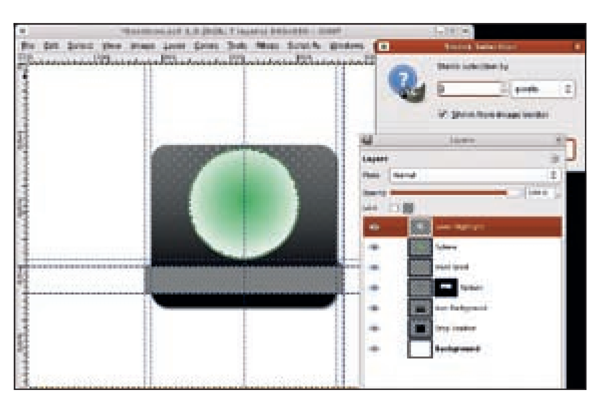
12 Цвета приложений

Сбросим цвета переднего плана и фона в значения по умолчанию, затем щелкнем по квадратику цвета переднего плана и установим RGB в 74, 164 и 75 соответственно. Теперь сменим цвет фона на значения RGB 28, 130 и 29. Закроем диалог выбора цвета. Оттенки зеленого будут использоваться в качестве градиента для иконки приложения.



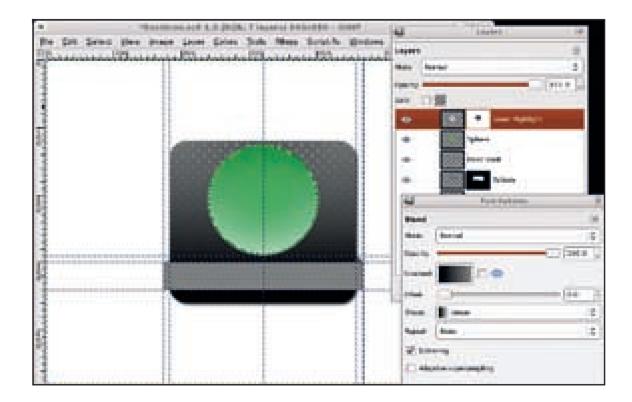
13 Сфера приложения

Добавим прозрачный слой на самый верх стопки слоев и назовем его «Сфера». Выберем овальное выделение и создадим круглое выделение в окне изображения, посреди иконки над выступом. В параметрах выберем заливку с радиальной формой и градиент Основной в фоновый. Протащим от центра к краю выделения.



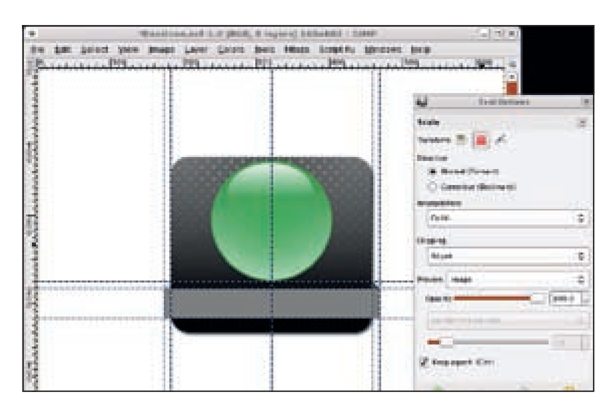
14 Нижний блик

Добавим еще один прозрачный слой поверх стопки слоев и назовем его «Нижний блик». Сожмем выделение на 2 пикселя, затем сбросим цвета в значения по умолчанию и инвертируем их. Используем инструмент Заливка с радиальной формой и градиент Основной в прозрачный с отмеченной галочкой опцией Развернуть, и протащим от центра к краю выделения.



15 Маска для нижнего блика

Добавим белую маску к слою «Нижний блик». Сбросим цвета переднего плана и фона в умолчания. Выберем заливку и выставим линейную форму и градиент «Основной в прозрачный». Протащим от верхней границы иконки вниз. Добавим прозрачный слой поверх стопки слоев и назовем его «Верхний блик».



16 Верхний блик

Выберем инструмент Масштаб с опцией «Преобразование: выделение». Протащим в окне изображения, чтобы изменить масштаб на 1/2 размера и примерно на 3/5 ширины. Сбросим цвета в значения по умолчанию и инвертируем их, затем сделаем заливку с линейной формой и градиентом «Основной в прозрачный» от верха выделения до середины.



17 Идентификатор приложения

Выберем инструмент Текст и шрифт с размером 160 и белым цветом. Наберем прописную G и поместим ее в центре окружности с помощью инструмента перемещения. Изменим режим слоя Верхний блик на Перекрытие. Более мелким шрифтом белого цвета наберем слово «GIMP» и расположим его поверх выступа, затем добавим отбрасываемую тень.



18 Название приложения

Сохраните это изображение как основу для иконок других приложений. Чтобы сохранить пиктограмму саму по себе, удалите слой белого фона и используйте инструмент Кадрирование для обрезки изображения, оставив только небольшую прозрачную рамку вокруг иконки. Сохраните изображение как **GimpIcon.png**. Используйте этот файл как базовую иконку при добавлении пиктограммы запуска приложения в *SimDock*. LXF

» **Через месяц** В завершение серии мы создадим принт для футболки.

DVDStyler: DVD

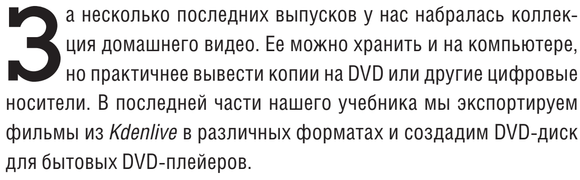
Часть 3: Отсняв очередной шедевр, **Энди «Вайнштайн» Ченнел** продемонстрирует, как раздать его родственникам и друзьям.





Наш эксперт

Энди Ченнел всю дорогу снимает автобиографию на видео – на старости лет он заново прокрутит свои звездные часы.



За несколько последних выпусков у нас набралась коллекция домашнего видео. Ее можно хранить и на компьютере, но практичнее вывести копии на DVD или другие цифровые носители. В последней части нашего учебника мы экспортируем фильмы из *Kdenlive* в различных форматах и создадим DVD-диск для бытовых DVD-плейеров.

Наряду с *Kdenlive*, которым мы пользовались на предыдущих двух уроках, мы применим *DVDStyler*, пакет с открытым исходным кодом, доступный для основных дистрибутивов Linux (а также Windows). На данный момент, последняя версия – 1.7.4, но мы возьмем версию 1.7.2: она включена в большинство дистрибутивов.

На первом этапе нужно экспортировать созданные нами фильмы из *Kdenlive*. *DVDStyler* управится и с несколькими форматами на одном диске, но «разводить зоопарк» нет нужды, если вы контролируете материал. Поэтому сначала нужно определиться с форматом. В какой-то мере это будет зависеть от разрешения, но помните, что видео в качестве 720p или 1080p не удастся проиграть на обычном телевизоре формата PAL. Для простоты мы будем экспортировать в стандартный формат, поэтому фоновые картинки для DVD-меню должны быть размером 720×576.



› DVD Wizard поможет вам наколдовать кино. Или еще что-нибудь.

› Обилие настроек вывода в *Kdenlive* посрамит не один коммерческий продукт.

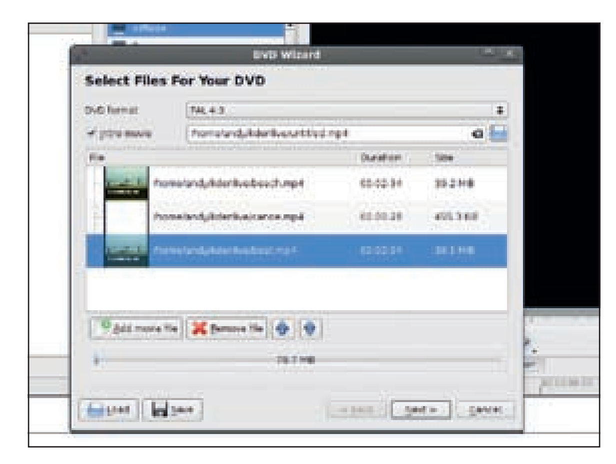
За дело!

Начнем с простого вывода для YouTube или любого другого видеохостинга: выберите в меню пункт Проект > Выполнить сборку [Project > Render], или нажмите Ctrl+Enter, чтобы посмотреть настройки. Их там довольно много, поэтому готовьтесь поработать. Прежде всего задается формат результирующего видеофайла. Возможные значения представлены в первом столбце – для выгрузки в Web укажите H.264. Во втором столбце, справа, появятся дополнительные настройки, задающие качество и степень сжатия файла. В общем случае, чем выше числа, тем лучше будет выглядеть результат. Но за это придется заплатить увеличением размера файлов.

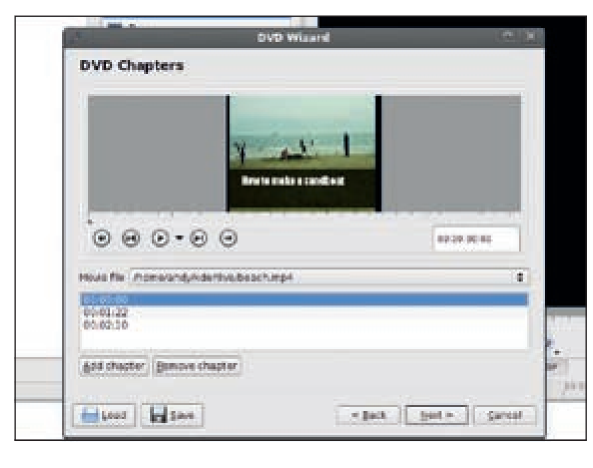
Мы всегда советуем выполнять сборку в два прохода: это позволяет проанализировать материал и выбрать оптимальный окончательный битрейт для всего файла, что приведет к лучшему результату. Однако однопроходное кодирование быстрее, и оно может быть удобным для тестовых пусков. Для первого вы-

» **Месяц назад** Мы украсили свой ролик эффектами *Kdenlive*.

для родни



› Выберите свои видеофайлы и расположите их в нужном порядке с помощью этих инструментов. Индикатор показывает, сколько места занято.

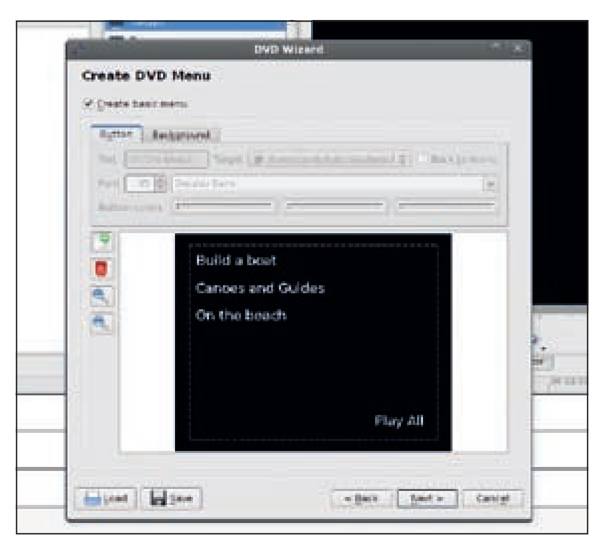


› Разделите фильм на части, чтобы зрителям было удобнее, особенно если он состоит из множества коротких клипов, как обычно в домашнем видео.

вода мы выбрали битрейт 1000 кбит/с и два прохода. Если результат нас не удовлетворит, вернемся назад и выберем большее значение.

Под настройками формата можно задать размер видео, экспортировать звуковую дорожку, добавить временной код – бегущие цифры в центре экрана, и при желании сделать так, чтобы фильм автоматически проигрался после окончания процесса. Здесь можно также указать, что именно выводить: весь проект, предварительно выделенную область или часть, ограниченную направляющими. После задания всех этих настроек, выберите каталог, где будет сохранен результат, и нажмите кнопку Сборка в файл [Render To File] в нижней части экрана.

Для вывода нашего 2,5-минутного фрагмента в двухпроходном режиме с битрейтом 1000 кбит/с понадобилось 6,35 с, и размер полученного файла был около 20 МБ. После увеличения битрейта до 2000 Кбит/с сборка в два прохода заставила подождать 8,02 с, а размер файла составил 39,4 МБ. За ходом процесса можно следить на вкладке Очередь заданий [Job Queue].



› Кнопки в меню можно назначить на отдельные фильмы или главы.

Благодаря модульной природе программы можно запустить процесс сборки в одном варианте, следом за ним – в другом, посмотреть «Вспомни, что будет» [«Вспомни, что будет» (англ. *FlashForward*) – телесериал, стартовавший в сентябре 2009 г. на канале ABC; в России транслируется на Первом канале, *– прим. пер.*], и вернуться к обоим фильмам, когда все будет готово. Полученные файлы H.264 (так же как Flash, MPEG2 и MPEG4) можно загрузить на YouTube безо всяких проблем.

Прожиг изнутри

Загрузить файл на сайт или отправить его по почте – это тоже здорово, но ничто не сравнится со сверкающим диском в руках. Мы рассмотрим два способа создания DVD: сначала только с *Kdenlive*, что даст нам весьма простой, но пригодный для воспроизведения продукт, а потом с *DVDStyler*, что позволит настроить внешний вид проекта.

Прежде чем создавать DVD, лучше закрыть все открытые в *Kdenlive* проекты: эта процедура сильно загружает процессор. Предварительно запасите все необходимое для создания диска: видеофайлы, фоновые картинки для меню и пустой DVD. Все подготовив, снова запустите *Kdenlive* и выберите пункт меню Файл > Мастер создания DVD [File > DVD Wizard]. Процессом управляет мастер, но он предоставляет достаточно возможностей, чтобы создать нечто уникальное.

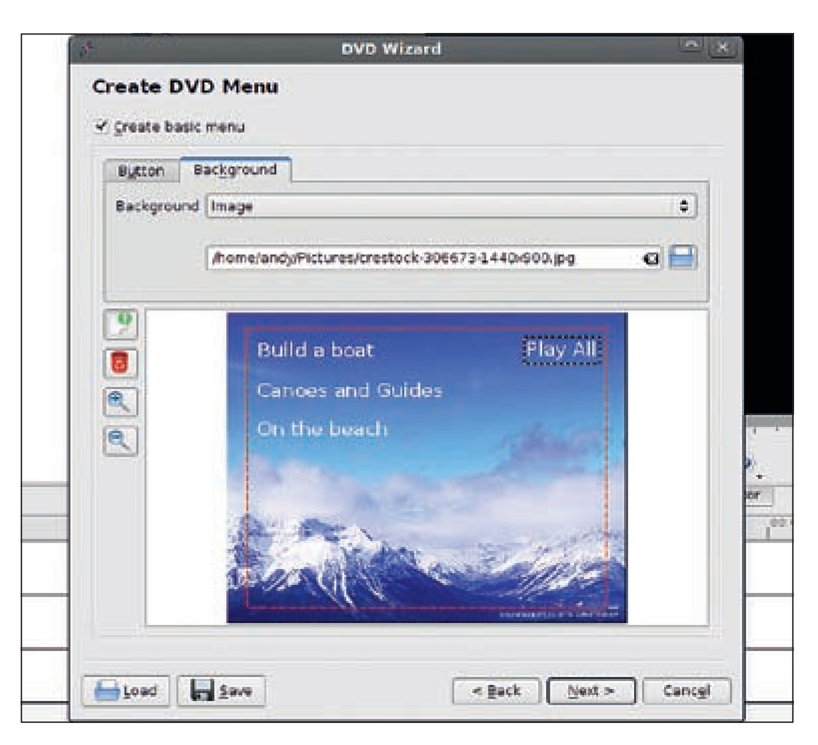
На первой странице задается общий формат DVD и добавляются видеофайлы. Формат указывается в выпадающем списке в верхней части окна; мы выбрали 4:3 PAL, потому что у нас камера со стандартным разрешением (не HD) и мы живем в Великобритании [для России такой вариант тоже подойдет, *– прим.ред.*].

Сразу под ними – Вступительный ролик [Intro Movie]. С помощью данного пункта можно задать ролик, который будет проигрываться сразу после вставки DVD в привод. В случае единственного фильма на диске можно пропустить остальную часть учебника, вплоть до записи. Однако предположим, что вы хотите добавить еще несколько. Если вы не добавили вступительный ролик, сразу после первой вставки диска в плейер на экране появится меню.

Чтобы добавить видеоклипы, воспользуйтесь кнопкой Добавить фильм [Add Movie File] для запуска менеджера файлов и выберите нужные клипы. После этого расположите клипы по порядку с помощью кнопок Вверх [Up] и Вниз [Down] или удалите ненужные с помощью кнопки Удалить файл [Remove File]. По ме- »



Если вы в итоге не хотите запутаться, создайте каталог **Final Cut**, где сохраните файлы результата – и только их.



› При выборе фонового изображения для проекта помните, что в мастере нельзя изменить его размер.

Скорая помощь

В нашей системе файловый браузер *Kdenlive* не показывал файлы MP4. Но мы смогли добавить их, набрав их имена в адресной строке.

ре добавления файлов индикатор будет показывать, сколько свободного места осталось на DVD. Покончив с добавлением файлов, нажмите Далее [Next].

На втором этапе создаются маркеры глав для всех фильмов – они позволят зрителю перемещаться по диску. В верхней части окна вы увидите свой клип с набором кнопок управления подачей и «считывающей головкой». Чтобы добавить маркер главы, перейдите на нужный момент фильма с помощью этих виджетов и нажмите на кнопку Добавить маркер [Add Chapter]. В списке под управлением подачей появится новая отметка; каждая из них идентифицируется только по времени. Для удаления маркера, выберите его и нажмите кнопку Удалить маркер [Remove Chapter]. Проделайте это для всех фильмов, выбирая их по очереди в выпадающем списке. Закончив, нажмите Далее [Next].

Следующий шаг – это меню. Если вас устраивает простой черный фон с белой кнопкой воспроизведения по центру экрана, можете перейти к следующему шагу. Но нам нужно нечто более изысканное, поэтому поставьте галочку Создать главное меню [Create Basic Menu] для просмотра дальнейших опций.

Создаем меню

По умолчанию в меню есть только одна кнопка Play [Воспроизвести]; нажав на нее, вы увидите, как все работает. Щелкните по ней и обратите внимание, что ее можно перемещать по экрану. На паре вкладок над окном предпросмотра можно определить кнопку и фон. Вы можете изменить текст на кнопке и связанное с ней действие; шрифт и его размер; а также цвета кнопки (активной, неактивной и при наведении курсора мыши). Измените их, воспользовавшись соответствующими виджетами. В левой части окна вы увидите инструменты для добавления кнопок в меню, удаления кнопок и изменения масштаба меню.

Пристроив все кнопки к месту, кликните по вкладке Фон [Background]. Здесь есть три варианта: Цвет [Colour], Картинка [Image] и Видеоролик [Video]. Измените цвет, нажав кнопку Цвет.

Добавьте картинку кнопкой Обзор [Browse], которая откроет окно выбора файлов. Помните: если вы создаете видео в стандартном разрешении, изображение должно быть размером 720×576 пикселей, и оно не должно накладываться на текст на кнопке или заслонять его. Последний вариант – добавление в меню видео. Пусть оно будет коротким: скажем, 20–30 секунд.

Снова нажмите Далее [Next]. Введите имя проекта, выберите временный каталог – обычно это **/tmp**, но помните, что временные файлы DVD могут быть огромными – потом нажмите кнопку Создать ISO-образ [Create ISO Image]. Программа подготовит образ, пригодный для записи на диск из *Brasero* или любой другой программы прожига DVD.

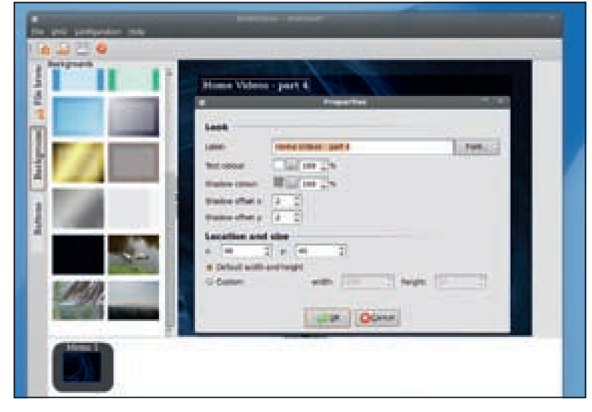
Учтите, что вам незачем делать все это одним махом. На самом деле, неплохо нажимать кнопку Сохранить [Save] в каждом окне мастера, чтобы сохранить уже введенные настройки, которые потом можно будет снова загрузить в мастер. Результат этих усилий – DVD.

Стили, вперед!

Конечно, мастер *Kdenlive* упрощает процесс, но результаты не особенно элегантны. Чтобы поправить дело, воспользуемся *DVDStyler*. После установки его можно будет найти в меню Приложения > Звук и видео. Интерфейс довольно разреженный, большинство настроек находятся на вкладках слева. В меню Настройка > Параметры [Configuration > Settings] убедитесь, что предлагаемые опции подходят для ваших фильмов. Формат видео должен быть таким же, а частота дискретизации звуковой дорожки скорее всего будет равна 48 кГц. Также можно задать длину главы по умолчанию – это упростит перемещение по фильмам кнопкой Пропустить [Skip].

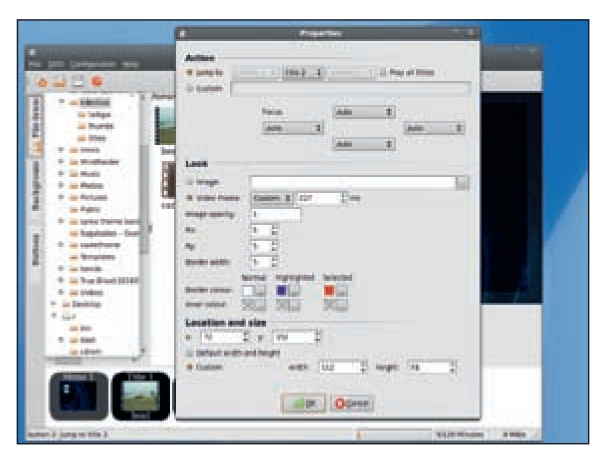
Главный вид приложения – это ваша сцена. Убедитесь, что кнопки и текст обведены тонким контуром, видимым на экране телевизора. Активная по умолчанию вкладка слева – Фоновые изображения [Backgrounds], и на сцену можно перетащить любое из установленных. Однако, кликнув правой кнопкой мыши в произвольном месте сцены и выбрав пункт меню Добавить > Изображение [Add > Image], можно открыть окно обозревателя и доба-

› В *DVDStyler* есть набор готовых фоновых изображений для меню, но легко добавить туда и собственное.



› Здесь можно изменить текст, его цвет и размер, а также точно задать положение.



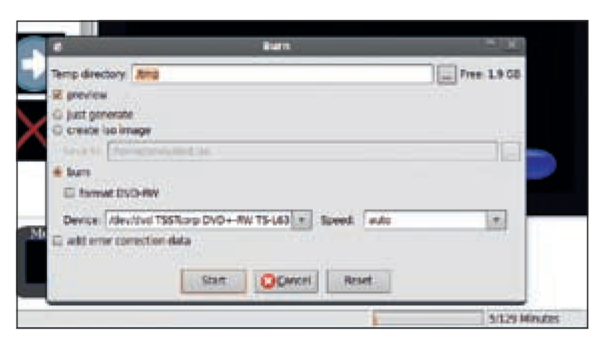


› Задайте для кнопок меню Свойства, включая Действие и Вид.

вить свои картинки. Вставив картинку в сцену, можно изменить ее размер, потянув за углы или стороны – правда, масштабирование непропорциональное, и довольно легко нечаянно перекосить изображение.

Теперь добавьте текст, кликнув правой кнопкой мыши на сцене и выбрав пункт меню Добавить > Текст [Add > Text] или Добавить > Текст с тенью [Add > Shadow Text]. В последнем случае текст будет с тенью, и это удобно, если цвет фона бледный, а шрифт традиционно белый (такой проще читать на экране телевизора). После выбора одного из данных вариантов появится диалоговое окно, где можно ввести какое-нибудь сообщение. Нажмите Enter, и текст появится на сцене; шрифт и кегль будут выбраны по умолчанию. Теперь дважды кликните на текстовом объекте, чтобы открыть окно Свойств [Properties], где можно изменить текст, шрифт (лучше всего выглядят гарнитуры семейства sans serif – без засечек) и цвета. Правый из двух цветовых квадратов – виджет, позволяющий задать прозрачность текста или тени, а два под ними определяют смещение тени. В нижней части окна можно задать размер и положение текста, но, наверное, проще перетащить его вручную.

Добавить на сцену материал можно двумя способами. Простейший – перейти на вкладку Файловый менеджер [File Browser], выбрать свои фильмы и перетащить их на сцену. Они появятся в виде кнопок с белым контуром и будут воспроизводиться при щелчке мышью или выборе с пульта. Двойной клик на любой из кнопок откроет окно Свойств. Выпадающие списки Фокус [Focus] соответствуют кнопкам Вверх, Вниз, Влево, Вправо пульта дистанционного управления DVD-плейера, и с их помощью можно установить порядок перехода для эффективной навигации по меню. У нас на экранном снимке выбрана кнопка Button 2; она находится под Button 1, поэтому в верхнем выпадающем списке следует выбрать Button 1. После этого в нижнем выпадающем списке можно указать Button 3.



› Параметры вывода DVD позволяет записать либо ISO-образ, либо сразу диск.



› Воспользуйтесь Свойствами, чтобы создать структуру меню DVD.

Проделайте это с остальными кнопками, создав таким образом линейный путь для навигации по содержимому.

В следующем разделе определяется внешний вид кнопки, и так как мы перетащили на сцену фильм, на кнопке будет показан кадр из него. Кадр можно заменить статичным изображением (нажмите на **...**, чтобы открыть обозреватель файлов) или выбрать Видеокадр [Video Frame], чтобы указать требуемый момент в фильме. Введите значение с клавиатуры, либо воспользуйтесь виджетом со стрелками, чтобы найти нужное.

Задайте прозрачность кнопки в виде числа между 0 (прозрачная) и 1 (непрозрачная) и укажите скругление углов кнопки с помощью значений Rx и Ry. Для квадратных углов установите значение 0. Значение Рамка [Border] определяет толщину окантовки, а ее цвет (если вам не по душе белый) можно изменить с помощью кнопки **...**.

Углубим

Замечательное свойство *DVDStyler* – ваши проекты могут быть настолько сложными или простыми, насколько вы этого хотите. Пожелав иметь страницу выбора сцен, можно создать новое меню (правая кнопка мыши > Добавить > Меню в селекторе страниц в нижней части экрана) и настроить кнопки, связанные с отдельными главами. Для этого добавьте кнопку, задайте действие (Action), как было описано ранее, и выберите подходящую картинку или кадр. После этого понадобится кнопка возврата в основное меню. Добавьте на сцену кнопку, дважды кликните по ней и выберите Menu 1 из выпадающего списка Действие. Тогда, щелкнув по ней, пользователь вернется в первое меню.

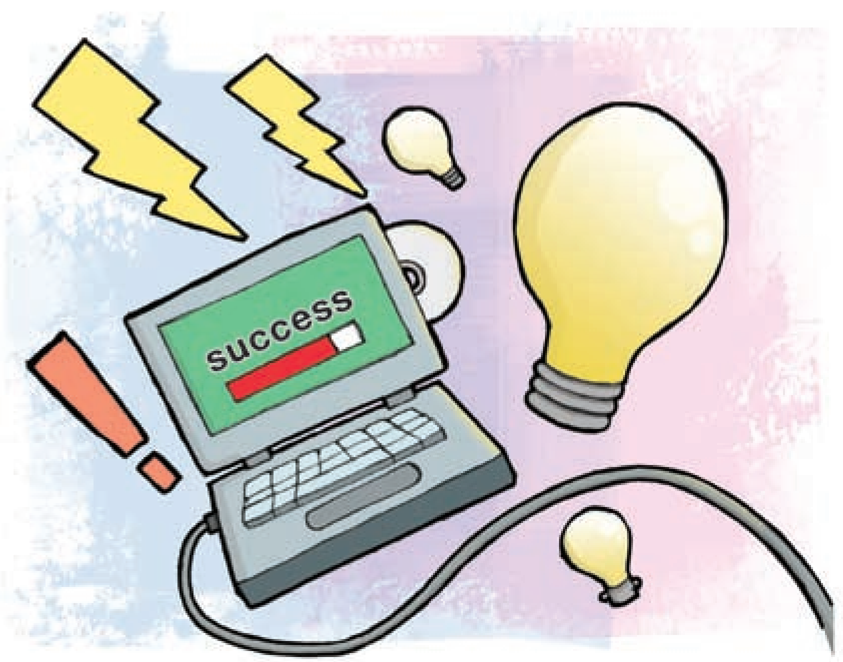
Таким способом можно создавать сложную структуру для самых изысканных проектов. У каждого меню может быть своя фоновая картинка или видео и собственная структура для пульта управления DVD.

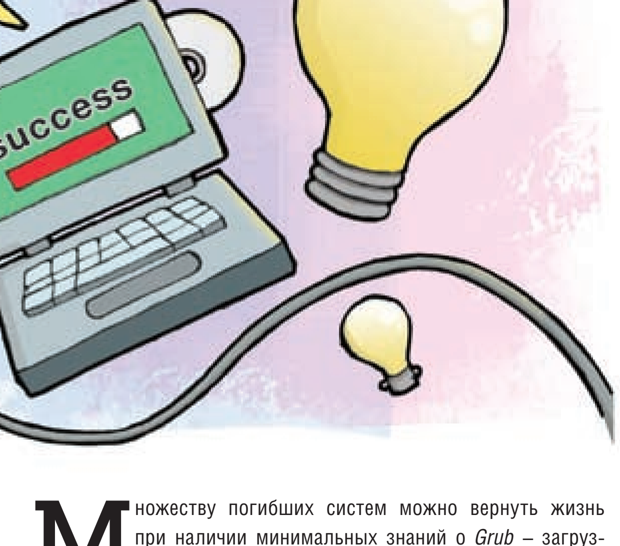
Удовлетворившись проектом, вставьте в привод чистый DVD и выберите пункт меню Файл > Прожиг DVD [File > Burn DVD]. Доступны различные варианты – в верхней половине окна можно «попросить» создать ISO-образ, чтобы записать его позже, или прожечь DVD-диск, включая возможность отформатировать DVD-RW перед записью. Нажмите кнопку Старт [Start], чтобы процесс пошел.

Как и многие творческие приложения, ради получения достойного результата *DVDStyler* требует времени и усилий. Однако ваши труды увенчаются достойной наградой – нет ничего приятнее, чем отправиться в прошлое вместе со своей семьей. LXF

Grub: Вникаем

Карл Флетчер вводит курс мультизагрузки в верное русло.







Наш эксперт

Карл Флетчер глобальный модератор на форумах openSUSE и рьяный фанат OpenSUSE и Fedora.

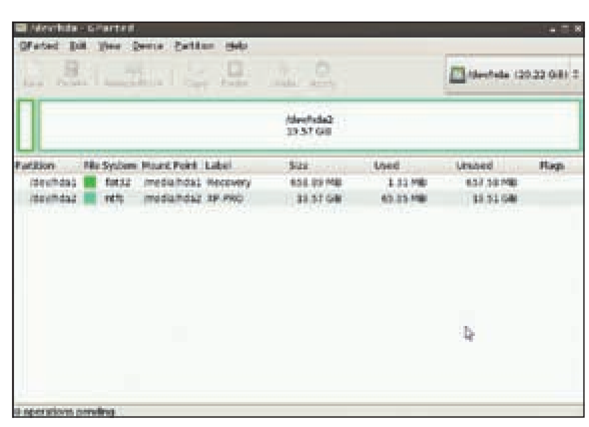
Множеству погибших систем можно вернуть жизнь при наличии минимальных знаний о *Grub* – загрузчике Linux; но трудности его освоения отпугивают новичков. Испортив файл настройки графического приложения, вы всегда откатитесь назад и исправите его, а вот поломка *Grub* оставит вас с незагружающейся машиной. Однако ответ найдется: это виртуализация. Воспользуемся магией *VirtualBox*, чтобы продвигаться по закоулкам *Grub* мелкими шажками, без риска повредить вашу систему, и по мере возможности постараемся обойтись без жутких письмен в командной строке. Займемся мы мультизагрузкой Vista/Windows 7 вместе с Linux, но принципы работы будут применимы независимо от используемой вами ОС.

Благодаря использованию *VirtualBox* можно делать сколько угодно ошибок, затем все сносить и начинать заново, если что-то пойдет не так. Как вы понимаете, применение метода проб и ошибок значительно более приятно, если вы уверены, что даже серьезные просчеты не будут иметь далеко идущих последствий. Все, что мы здесь проделаем, будет выполняться в виртуальной машине на хосте Linux, но применимо и в вашей реальной Linux-машине. (Данный урок предполагает умение работать с *VirtualBox*.)

На нашем уроке мы также разберем самую популярную версию *Grub*, известную как Grub legacy. Некоторые новые дистрибутивы – к примеру, Ubuntu 9.10 – уже идут со следующей версией *Grub*, именуемой Grub 2; его мы столь же детально осветим в следующем выпуске.

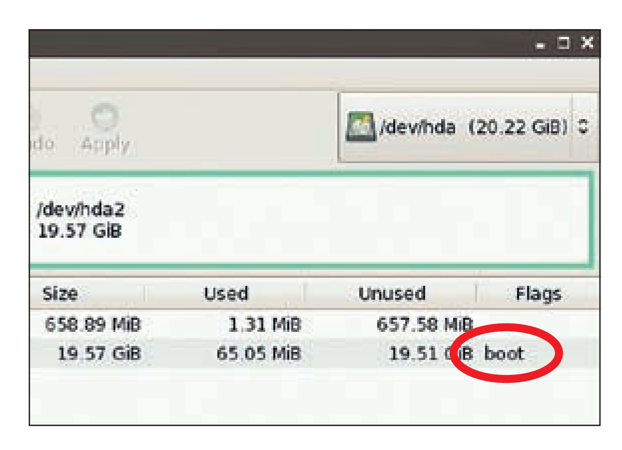
Виртуальные разделы

Создайте в *VirtualBox* фиксированный диск как минимум на 20 ГБ. Чтобы двигаться дальше запишите ISO-образ Parted Magic на CD-R (с http://partedmagic.com), выберите наш диск и создайте два раздела. Первый – hda1 – должен быть разделом FAT32 объемом 660 МБ, а второй – занимать оставшееся место и иметь тип NTFS. Подготовив почву для установки Windows, вы можете заметить отсутствие под рукой подходящего диска. Ничего страшного – создайте разделы и оставьте их пустыми, или попрактикуйтесь с мультизагрузкой двух дистрибутивов Linux. Мы пытаемся эмулировать реальное окружение, и нельзя не признать, что компьютеры с предустановленной Windows продаются чаще, чем без нее; так зачем бы ее удалять? На этом уроке установкой Windows я воссоздаю вполне типовую конфигурацию, и вы можете делать так же, если захотите. В нашем случае я использовал XP, но Vista или Windows 7 тоже подойдут. В зависимости от раздела, куда установлена Windows, вы отметите только одно маленькое отличие – флаг boot. Помните о нем: это важно.



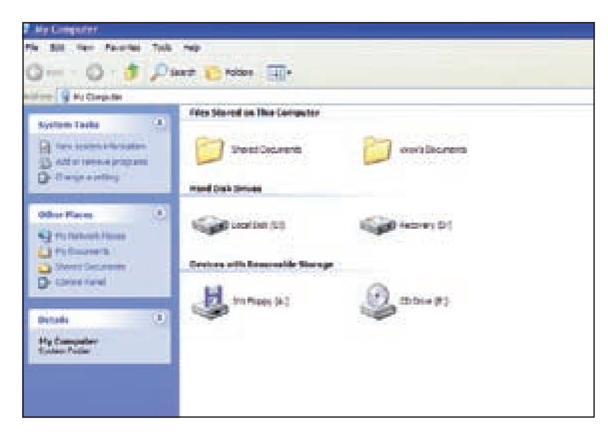
› Компьютеры с Windows обычно имеют как минимум два раздела, но раздел восстановления hda1 — просто пустышка.

Обратим также внимание на то, как Linux называет разделы. В нашем случае он обозначит их **hda** плюс номер, но иногда это бывает **sda**. Чтобы не отклоняться от темы, просто скажем, что разделы разбивают ваш жесткий диск на части, как если бы было множество маленьких жестких дисков вместо одного.

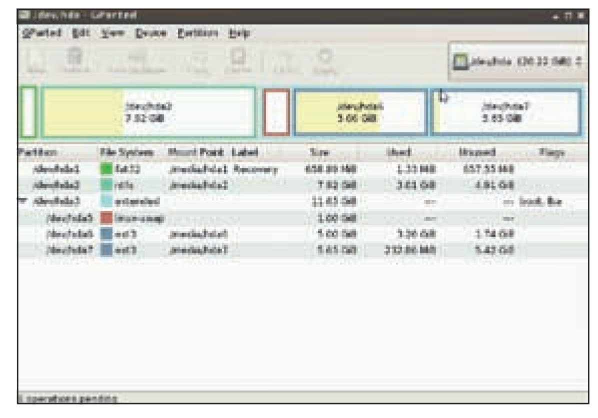


› При установке ОС обратите внимание на флаг boot.

в загрузку



› Вид в Windows: так XP видит разделы.



› Вид после установки из Parted Magic.

На экранном снимке вверху видно, что **Recovery D:** расположен перед **XP Local Disk C:** в таблице разделов, хотя и обозначен буквой, которая стоит в алфавите дальше. Если вы пользователь Windows, перестаньте думать о ваших разделах «виндовато» – использование букв дисков очень неинформативно и может вызвать проблемы в процессе установки Linux, который нумерует разделы числами и по порядку: sda1, sda2 или hda1, hda2.

Итак, наши цели ясны: мы хотим установить дистрибутив Linux и использовать *Grub* для управления загрузкой и Linux, и Windows. Ради доступа к богатым возможностям настройки, я выбрал OpenSUSE, которая высказывает свои соображения по инсталляции в виде, представленном на изображении внизу справа.

Процесс установки

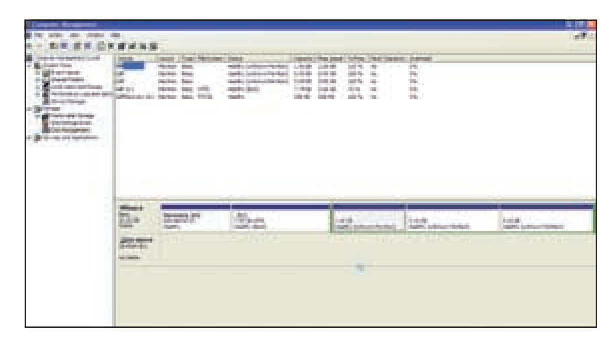
Заметьте, что в позиции Booting приглашения к установке, *Grub* помещен в **sda3** (расширенный раздел). Благодаря этому загрузчик Windows остается неизменным. Однако вы сразу же и заметите, что флаг загрузки, на который мы обращали внимание ранее, переместился с **sda2** на **sda3**. Иногда инсталлятор с этим не вполне справляется. Но в ходе тестовых испытаний при написании этой статьи установщик правильно поместил *Grub* в **sda3**, а также успешно включил опцию Windows в загрузочном меню. (Большинство дистрибутивов делают то же самое, хотя можно записать *Grub* в Master Boot Record – первую часть жесткого диска, где хранится информация о загрузке. Тогда флаг загрузки должен быть помещен в корневой раздел OpenSUSE, **sda6**). Изображение вверху справа показывает порядок разделов после окончания установки ди-

> «Grub помещен в sda3; тогда не изменится код загрузки Windows.»



Скорая помощь

Используйте в *VirtualBox* ISO-образы: это снимает необходимость прожига CD/DVD и увеличивает производительность.



› Соображения по установке в Windows…

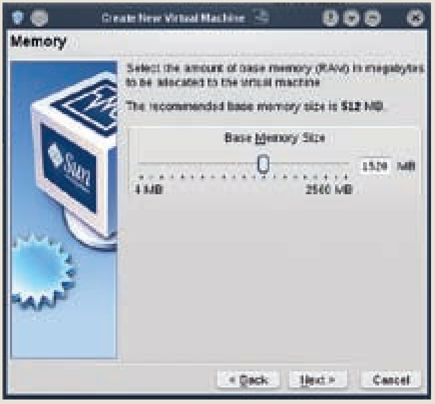


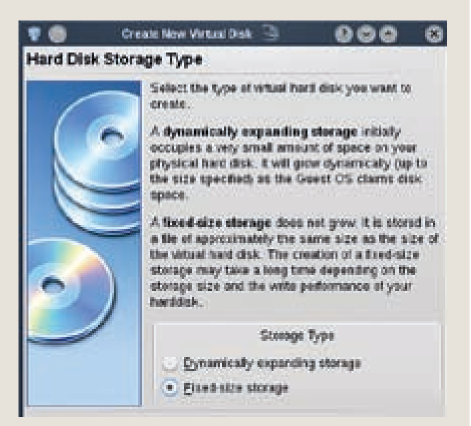
› …и их эквивалент в более информативном окне OpenSUSE.

» **Пропустили номер?** Узнайте на с. 107, как получить его прямо сейчас.

Шаг за шагом: Создаем виртуальный ПК в VirtualBox







1 Выбираем тип ОС

Дайте виртуальной машине имя и выберите тип ОС для установки (Windows или Linux).

2 Задаем память

Перемещением ползунка установите доступную виртуальной машине память (она не должна превышать половины объема вашего реального ОЗУ).

3 Монтируем ISO-образ

Создайте образ виртуального жесткого диска достаточного размера для целевой ОС, затем укажите *VirtualBox* ISO-образ дистрибутива в настройках CD-ROM.

стрибутива. Отметим что флаг загрузки теперь установлен на **hda3**, а весь объем **hda3** и содержащихся в нем логических разделов получен урезанием раздела **hda2**. Да, расширенный раздел – это, по сути, всего лишь контейнер для логических разделов, которыми являются **hda5** и **hda6**. Расширенные разделы помогают обойти исторический лимит в четыре основных раздела. Не лишним будет напомнить, что перед любой установкой Linux вы должны сделать резервную копию данных и выполнить дефрагментацию Windows.

Как уже говорилось, в этом примере *Grub* расположен на **hda3** и установщик переключил флаг boot с **hda2** на **hda3**. Поэтому при загрузке компьютер смотрит на **hda3**, и затем появляется меню *Grub*. Теперь, если вы загрузите *Parted Magic* и переведете флаг boot назад на **hda2**, при перезагрузке машина будет запускать Windows без меню *Grub*. Linux, конечно, никуда не денется: вы увидите его разделы в Управлении компьютером Windows. Однако возврат флага boot снова на **hda3** не восстановит *Grub*; этого не произойдет и после переустановки Windows. Чтобы вернуть *Grub*, нужно выполнить его восстановление вручную – давайте попробуем сделать это сейчас.

Скорая помощь

Parted Magic показывает разделы как **hda**, но OpenSUSE называет их **sda**.

Восстановим Grub

Дистрибутивный диск OpenSUSE можно применить для восстановления *Grub* через интерфейс инсталлятора. Я решил показать вам другой подход, с помощью командной строки. Если у вас есть LiveCD OpenSUSE (а он у вас есть – и это **LXF**DVD!), загрузитесь с него, затем откройте окно консоли и введите **su** – вы попадете в командную строку с правами root и будете готовы к действиям.

Grub с Vista или Windows 7

В случае установки на один жесткий диск возможна загрузка Vista или Windows 7 через *Grub*. Но сервисные пакеты Windows Vista и Windows7 могут не установиться (как произошло при моих экспериментах). Вы можете обойти эту проблему, временно возвратив загрузчик Windows и флаг boot перед обновлением Windows, а затем восстановив *Grub*. Если Linux и Windows 7 установлены на разных дисках, лучшим решением будет оставить MBR на жестком диске Windows нетронутым и установить *Grub* на другой жесткий диск, с записью 'chainloader' в *Grub*, передающей управление загрузчику Windows.

Попав в командную строку root, ищите раздел, содержащий загрузчик OpenSUSE. Затем переустановите *Grub* со ссылкой на этот раздел. Сперва найдем установку OpenSUSE:

```
grub
```

Компьютер вернет приглашение **grub>**, показав вам, что вы находитесь в командной строке *Grub*. Теперь введите

```
find /boot/grub/menu.lst
```

– и компьютер должен вернуть **(hd0,4)**.

Здесь **(hd0,4)** – это указатель *Grub* на мою установку SUSE. При следовании нашему уроку ваш указатель должен быть похожим, но в реальной ситуации может отличаться, поэтому при необходимости замените его на ваше правильное значение. Теперь, имея указатель, продолжите так:

```
root (hd0,4)
Filesystem type is ext2fs, partition type 0x83
setup (hd0)
Checking if /boot/grub/stage1 exists ... yes:
```

Ваш компьютер должен в итоге вернуть вот что:

```
Succeeded.......Done
quit
reboot
```

Компьютер должен перегрузиться и предстать пред вами с загрузочным меню *Grub*, из которого вы можете попасть в OpenSUSE. Приведенные здесь команды будут такими же и в других дистрибутивах Linux.

О Grub и нескольких дисках

Это наиболее вероятный сценарий для настольных ПК, где не редкость наличие двух внутренних жестких дисков. Вообще-то более идеального стечения обстоятельств и быть не может, поскольку это позволяет выделить основную загрузочную запись (MBR) одного из дисков для *Grub*. Если у вас два жестких диска и вы намерены использовать Windows, отведите под Windows один из них. Windows установит свой загрузчик в MBR этого диска. Если у вас Windows-машина, в которую вы добавляете новый жесткий диск, выгоднее будет оставить исходный диск в покое, а все новое установить на второй диск. Ключ к этому – в вашем BIOS. Вы должны настроить его так, чтобы новый диск был первым при загрузке.

Рекомендуем выполнять установку вручную, не полагаясь на параметры по умолчанию, и самостоятельно разбить новый диск, установить точки монтирования для диска с Windows и разместить *Grub* в MBR нового жесткого диска. Возможно, что установщик Linux добавит неверный код ссылки на загрузчик Windows в меню *Grub*. Паниковать тут нечего – вы все еще можете загрузиться в Windows, если надо, временно изменив настройки BIOS и снова установив жесткий диск Windows первым, как загрузочный. Но мы пойдем другим путем – мы хотим использовать *Grub*, и поэтому должны поправить файл **menu.lst**, расположенный обычно в **/boot/grub/menu.lst**. Если у вас два жестких диска, **menu.lst** может выглядеть примерно как здесь, где Win XP расположена на первом разделе второго жесткого диска. Файл включает *map*, функцию отображения *Grub*, которая исправляет ситуацию для **boot.ini** 2000/XP.

```
title WIN XP
map (hd0) (hd1)
map (hd1) (hd0)
rootnoverify (hd1,0)
chainloader +1
```

или

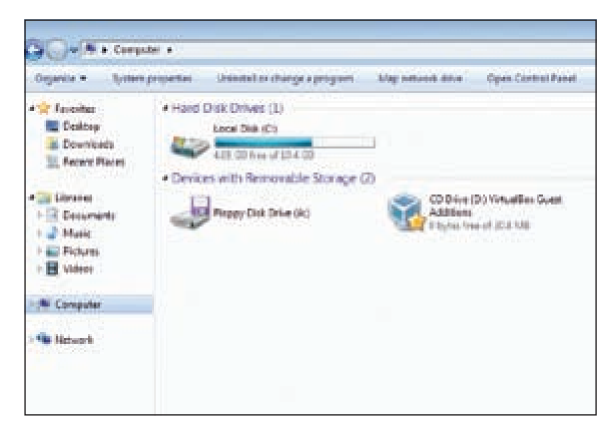
```
title WIN XP
map (hd0) (hd1)
map (hd1) (hd0)
rootnoverify (hd1,0)
chainloader (hd1,0)+1
```

Объясним причину использования здесь функции **map**. Фрагмент **rdisk(0)partition(1)** означает раздел 1 диска 0. Но сделанные изменения привели к тому, что раздел стал 1 на диске 1. Поэтому мы используем функцию *map* для трюка с ОС, чтобы видеть физический диск 1 как фиктивный диск 0 и только при загрузке XP.

Теперь фишка вот в чем. Vista использует другой загрузчик, не содержащий фрагмент **rdisk(x)partiton(y)**, и ей не нужна функция **map**. Она ее игнорирует, загружается и прекрасно работает без нее, потому что изменение расположения не влияет на новый загрузчик в Vista.

Если вы хотите гарантированно узнать, где загрузочный раздел, поищите тот, на котором содержатся файлы загрузчика: в Windows 2000/XP это **ntldr**, **ntdetect.com** и **boot.ini**. В Vista и Windows 7 видимый файл загрузчика – **bootmgr** и директория **boot**. Windows Vista и Windows 7 не нуждаются в функции **map**.

> «Красота VirtualBox в том, что можно превзойти свой уровень.»



› **Надеюсь, вы доберетесь досюда. Любите ли вы ее или ненавидите, но это Windows 7, которой бредят множество людей.**

Зачем все эти Windows?

Вы можете подумать, что мы слишком много носимся с Windows, но это голый прагматизм: большая часть вопросов, встречавшихся мне на форумах OpenSUSE, касалась мультизагрузки и проблем с ней при использовании Linux и Windows. Это обычно основной сценарий проблем *Grub*, потому мы и уделили ему столько внимания. Очень часто начинающие пользователи в этом пункте попадают в беду. Им иногда кажется, что все потеряно и их машина окончательно погибла! Но такое, хоть и бывает, обычно очень маловероятно. На этих четырех страницах мы управляли *Grub* и чинили его, разбивали жесткий диск, устанавливали ОС, переключали флаги загрузки и близко познакомились с требованиями загрузки Windows. Конечно, не забывайте, что все это было в *VirtualBox*.

Наконец, экстремалы в дополнение могут попробовать устроить себе «встряску». Если у вас установлены и работают Windows и Linux, как на данном уроке, загрузите *Parted Magic*, смотрируйте раздел Windows и удалите файлы загрузчика (**ntldr** и **bootmgr**). Затем попытайтесь загрузить Windows из *Grub* – естественно, будет сбой. Теперь попрактикуйтесь в исправлении загрузки Windows, а когда закончите – переустановите *Grub*.

Это означает, что Vista будет на первом жестком диске (HD1), и в первом разделе будет такой код:

```
title Vista
rootnoverify (hd0,0)
chainloader (hd0,0)+1
```

А для Vista на втором жестком диске и первом разделе – такой:

```
title Vista
rootnoverify (hd1,0)
chainloader +1
```

Более подробную информацию вы найдете, набрав **info grub** или **man grub** в терминале или посетив www.gnu.org/software/grub/manual/grub.html. Не помешает также заглянуть на сайт www.supergrubdisk.org.

Взгляд на установку Windows 7

Наконец, пройдемся по установке Windows 7. Снова, как и ранее, можете скопировать этот пример в *VirtualBox*.

Загрузившись с LiveCD OpenSUSE KDE 4 и запустив инсталлятор, вы увидите предложение урезать раздел Windows и создать расширенный раздел и три логических: подкачки, корневой и домашний.

В части Booting приглашения к установке инсталлятор предложит установить *Grub* в расширенный раздел **sda2** и включит Windows-строку в меню. Позже мы увидим, что в результате установки флаг boot переключен на **sda2**. Так как Windows-раздел был урезан, запустится проверка CHKDSK. Это займет около 20 минут, в зависимости от размера раздела и доступного свободного места. *Grub* поместит в **/boot/grub/menu.lst** следующий код:

```
title Windows
rootnoverify (hd0,0)
chainloader +1
```

Наводим порядок

Закончив, можете освободить 20 ГБ простым щелчком на Удалить в *VirtualBox*. Красота тестирования ваших задумок в *VirtualBox* в том, что вы можете превзойти свой обычный уровень компетенции. Это великолепный способ повышения квалификации. Можно было бы, наверное, сравнить это с авиатренажером – он очень реалистичен, но если вы войдете в штопор, то это не смертельно. Свершив такие подвиги, можете заслуженно похлопать себя по спинке! LXF

Скорая помощь

Используйте виртуальную машину для практики в установке и знакомства с этим процессом. Вы можете придумать любой сценарий и проиграть его в реальности. Я всегда создаю диск фиксированного размера – он работает намного лучше.



Скорая помощь

Так как это новая установка, нет нужды делать дефрагментацию Windows.

OpenOffice.org:

Дмитрий Попов расскажет, как создать и обработать формы в *OpenOffice.org*, а также как сохранить добытые данные в виде текста или в базе данных Base.



Наш эксперт

Дмитрий Попов пишет о Linux и открытом ПО уже несколько лет. Он автор популярного расширения *Writer's Tools* для *OpenOffice.org* и рьяный сторонник открытых офисных приложений.

OpenOffice.org Writer – это не только солидный текстовый редактор. С использованием его возможностей дизайнера форм можно легко генерировать анкеты любого вида, от простых тестов до продвинутых опросников. И хотя создание форм во *Writer* не отличается сложностью, их обработка и извлечение данных из них – это совсем другое дело. К счастью, *OOo* сам предлагает инструменты для создания простого, но эффективного решения по обработке форм. С помощью *OpenOffice.org Basic* можно написать макрос, обрабатывающий форму, а извлеченные данные сохранить в текстовом файле или в базе данных *OpenOffice.org Base*. Изучая это решение обработки форм, вы узнаете несколько продвинутых технологий *OpenOffice.org Basic*, способных пригодиться вам в других проектах.

Любой бизнес первым делом требует формы сбора данных. Начнем с простой, для введения имен и фамилий пользователей. Создайте новый документ *Writer* и выберите Вид > Панели инструментов > Дизайн формы, чтобы появилась панель управления Дизайн формы. Нажмите на Режим проектирования, затем выберите кнопку Текстовое поле и нарисуйте в документе текстовое поле. Дважды щелкните на только что созданном поле, чтобы открылась палитра Свойства, и введите в поле Имя FirstName. С помощью других опций этой палитры можно настроить свойства поля: например, его внешний вид, шрифт и цвет рамки. Таким же образом нарисуйте другое текстовое поле под названием LastName. Сделав это, сохраните полученную форму.

Теперь начнем работать над макросом. Чтобы все было сделано аккуратно, создадим для наших экспериментов отдельную библиотеку. Зайдите в Сервис > Макросы > Управление макросами > OpenOffice.org Basic, нажмите кнопку Управление, переключитесь в раздел Библиотеки и создайте новую библиотеку под названием ProcessForm. Нажмите кнопку Изменить, и откроется редактор *OOo Basic*. Начнем с написания простого макроса, который инициализирует форму, считывает данные из ее полей и отображает извлеченные значения во всплывающем сообщении.

Создание формы

Для инициализации формы нужны два оператора. Первый велит макросу использовать текущий открытый документ:

```
Doc=ThisComponent
```

Второй оператор отыскивает в документе определенную форму по ее индексу. В нашем случае в документе есть только одна форма с индексом, равным нулю, и оператор, инициализирующий переменную Form, имеет вид

```
Form=Doc.Drawpage.forms.getbyIndex(0)
```

Извлечение данных из формы – тоже двухшаговый процесс. Во-первых, макрос ищет поле по его имени, используя метод **getByName**:

```
TextBox=Form.getByName("FirstName")
```

Найдя поле, макрос извлекает из него текстовые данные следующим оператором:

```
FirstName=TextBox.String
```

Точно так же можно взять данные из поля LastName:

```
TextBox2=Form.getByName("LastName")
LastName=TextBox2.String
```

Наконец, чтобы убедиться в правильности работы макроса, добавим оператор, отображающий собранные данные в окне сообщения:

```
MsgBox FirstName & " " & LastName
```

Вот полный листинг макроса, который нужно ввести в редакторе *OOo*:

```
Sub FetchFormData
Doc=ThisComponent
Form=Doc.Drawpage.forms.getbyIndex(0)
TextBox1=Form.getByName("FirstName")
FirstName=TextBox1.String
TextBox2=Form.getByName("LastName")
LastName=TextBox2.String
MsgBox FirstName & " " & LastName
End Sub
```

После ввода макроса перейдите в документ, содержащий форму, заполните два поля и сохраните форму. Чтобы запустить макрос, щелкните Сервис > Выполнить макрос... и выберите макрос FetchFormData из библиотеки ProcessForm. Если все работает правильно, вы увидите окно сообщения с данными из полей формы.

Отображение извлеченных данных в окне сообщения демонстрирует нам, что макрос работает; но польза от него пока невелика. Давайте изменим его так, чтобы он записывал значения через

Все по форме

запятую (CSV) в текстовый файл, который затем можно импортировать и обработать в *Calc* или в другом приложении:

```
Sub WriteToFile()
Doc=ThisComponent
Form=Doc.Drawpage.forms.getbyIndex(0)
TextBox=Form.getByName("FirstName")
FirstName=TextBox.String
TextBox1 = Form.getByName("LastName")
LastName= TextBox1.String
FilePath="/home/user/formdata.csv"
f1=FreeFile()
Open FilePath For Append Access Read Write Lock Write As #f1
Print #f1, FirstName + ", " + LastName + Chr(13)
Close #f1
End Sub
```

Определение путей к файлам

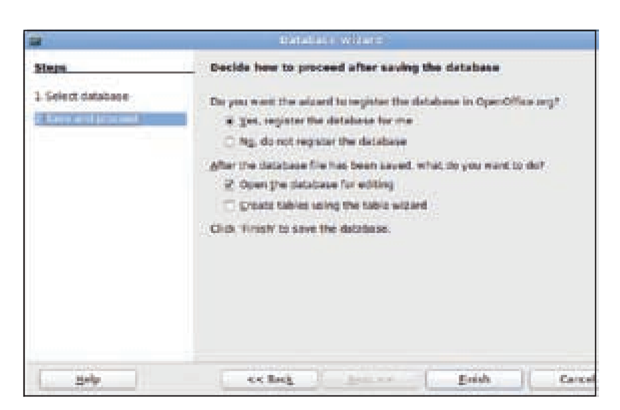
Та часть макроса, что записывает извлеченные данные в текстовый файл, начинается с оператора **FilePath="/home/user/formdata.csv"**. Как вы, наверное, уже догадались, этот оператор определяет целевой текстовый файл. Вместо того, чтобы прописывать путь к файлу в макросе вручную, воспользуйтесь процедурой **InputBox** — пусть его определит пользователь:

```
FilePath=InputBox("Введите путь к файлу", "Внимание")
```

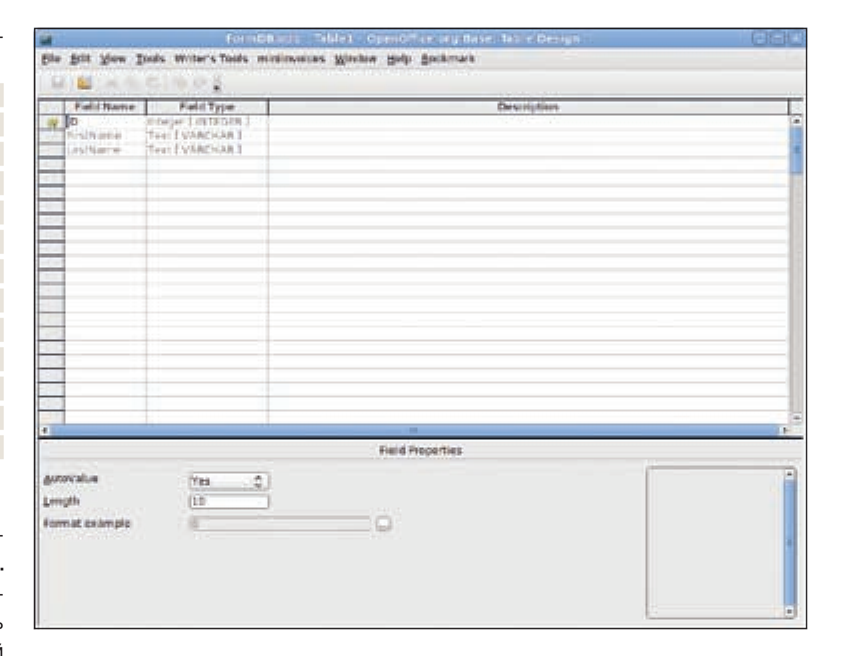
Альтернатива – обратиться к службе **com.sun.star.util.PathSubstitution**, чтобы узнать путь к каталогу настроек пользователя *OOo* (например, /home/user/.openoffice.org/3/user) и применить его в качестве местоположения текстового файла. Как видно из названия, эта служба заменяет определенную переменную на путь. В нашем случае мы используем переменную **$(user)**, и служба возвратит путь к каталогу настроек пользователя:

```
SubstService=CreateUnoService("com.sun.star.util.
PathSubstitution")
UserPath=SubstService.substituteVariables("$(user)", true)
```

Теперь используем полученное имя каталога для задания имени файла:



› При создании базы данных выберите опцию Да, зарегистрировать базу данных — в дальнейшем это сэкономит ваше время.



› Чтобы хранить данные, извлеченные из формы, нужно создать простую таблицу. В Свойствах задайте длину всех полей.

```
FilePath=UserPath + "/formdata.csv"
```

Неважно, какой из этих вариантов вы выберете: способ, которым макрос будет записывать данные, останется тем же. Чтобы открыть файл, макрос определяет уникальное число, часто называемое номером файла или каналом данных. Это делается с помощью функции **FreeFile**, которая возвращает номер файла для использования в макросе:

```
f1=FreeFile()
```

Затем макрос открывает файл командой **Open**:

```
Open FilePath For Append Access Read Write As #f1
```

Режим **For Append** создает новый файл или же открывает существующий и помещает курсор в его конец; иными словами, наш макрос будет дописывать данные. Режим **Access** определяет права доступа; в нашем случае макрос может читать и изменять файл. Чтобы ограничить доступ к файлу на время его использования, макрос применяет режим **Lock**. Когда текстовый файл открыт, макрос записывает в него данные из полей формы и закрывает файл:

```
Print #f1, FirstName + ", " + LastName + Chr(13)
Close #f1
```

Обратите внимание на **Chr(13)** в конце оператора **Print**. Это символ возврата каретки, чтобы следующая строка, которую будет записывать макрос, начиналась с новой строки.

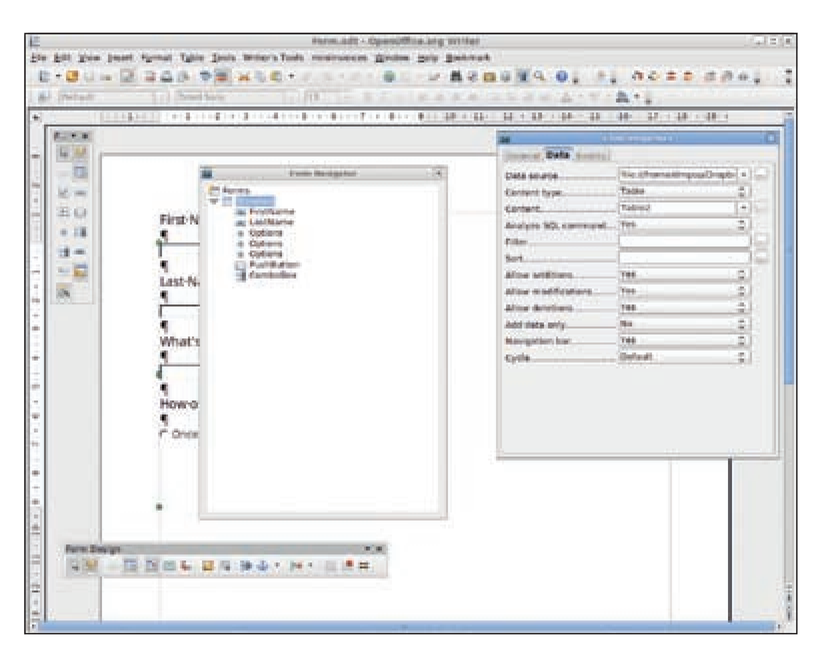
Сохранение данных в базе

Данные в текстовом файле можно импортировать и обрабатывать любой программой, поддерживающей формат CSV, но есть и возможность сохранять обработанные данные непосредственно в базе, минуя стадию промежуточных файлов. Для этого зайдите в Файл > Создать > Базу данных..., выберите пункт Создать »



Скорая помощь

Чтобы скопировать данные из базы в электронную таблицу *Calc*, щелкните правой кнопкой по нужной таблице и выберите Копировать. Переключитесь в *Calc*, выберите Правка > Вставить. И все!



› Для добавления выпадающего списка или еще чего-нибудь надо связать вашу форму с базой данных.

новую базу данных и нажмите Далее. Выберите Да, зарегистрируйте базу данных и отметьте Открыть базу для редактирования, затем нажмите на кнопку Готово. Сохраните созданную базу данных под именем **FormDB.odb**. Перейдите в раздел Таблицы и нажмите на Создать таблицу в режиме дизайна.

В таблицу следует внести три поля. Начнем с ID, оно будет ключевым полем таблицы. Введите **ID** в столбце Название поля, а в выпадающем списке Тип поля выберите Целое [Integer]. Щелкните правой кнопкой мыши по зеленому маркеру рядом с именем поля и выберите опцию Первичный ключ. Наконец, установите опцию Автозначение на Да и введите **10** в поле Длина. Добавьте поля FirstName и LastName, установите им тип Текст [Varchar] и задайте для каждого длину. Сохраните полученную таблицу как **Table1**, и ваша новая база данных готова.

> «Можете оживить вашу форму, включив в нее списки и опции.»

Вы уже знаете, как извлекать данные из полей формы; осталось написать код, сохраняющий полученные значения в базе данных. Первым делом макрос должен установить связь с базой данных **FormDB.odb**, используя службу **com.sun.star.sdb.DatabaseContext**:

```
DBContext=createUnoService("com.sun.star.sdb.
DatabaseContext")
DataSource=DBContext.getByName("FormDB")
DB=DataSource.GetConnection ("","")
```

Чтобы этот код заработал, база данных **FormDB.odb** должна быть зарегистрирована *OOo* в качестве источника данных. Если, создавая базу, вы выбрали опцию Да, зарегистрировать базу данных, то вам ничего больше делать не нужно; а если нет, потребуется зарегистрировать базу данных вручную. В *OOo* зайдите в Сервис > Параметры, там выберите раздел База данных OpenOffice.org > Базы данных. Нажмите Создать..., выберите файл **FormDB.odb**, убедитесь, что в поле Зарегистрированное название стоит FormDB, и нажмите OK.

Для управления данными, содержащимися в *Base*, *OOo Basic* применяет SQL-запросы, и для сохранения данных формы в таблицу Table1 базы данных FormDB понадобится команда SQL **INSERT INTO**. Запрос имеет следующий формат:

```
INSERT INTO таблица ("Поле1", "Поле2") VALUES
('Значение1','Значение2')
```

Вот как должен выглядеть оператор, вставляющий значения FirstName и LastName в поля таблицы Table1:

```
SQLQuery="INSERT INTO ""Table1"" (""FirstName"",
""LastName"") VALUES " + "('" + FirstName + "','" + LastName +
"')"
```

Наконец, макрос выполняет указанный SQL-запрос и закрывает соединение с базой данных:

```
SQLStatement=DB.createStatement
Result=SQLStatement.executeQuery (SQLQuery)
DB.close
DB.dispose()
```

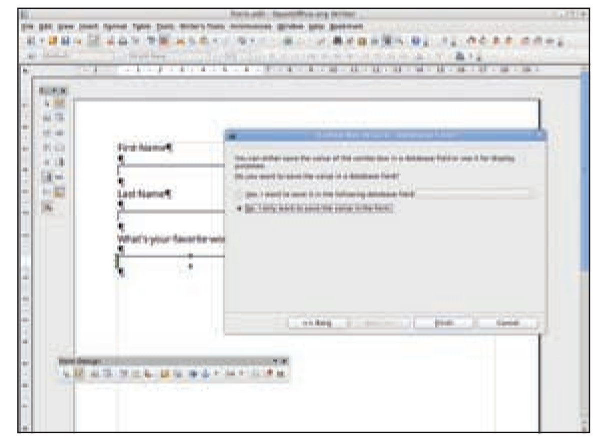
Вот полный текст макроса для ознакомления:

```
Sub SaveInDB()
Doc=ThisComponent
Form=Doc.Drawpage.forms.getbyIndex(0)
TextBox1=Form.getByName("FirstName")
FirstName=TextBox1.String
TextBox2=Form.getByName("LastName")
LastName=TextBox2.String
DBContext=createUnoService("com.sun.star.sdb.
DatabaseContext")
DataSource=DBContext.getByName("Basket")
ConnectToDB=DataSource.GetConnection ("","")
SQLQuery="INSERT INTO ""Table1"" (""FirstName"",
""LastName"") VALUES " + " ('" +
FirstName + "','" + LastName + "')"
SQLStatement=ConnectToDB.createStatement
Result=SQLStatement.executeQuery (SQLQuery)
ConnectToDB.close
ConnectToDB.dispose()
End Sub
```

Выпадающие списки

Теперь, зная, как извлекать данные из текстовых полей, можете оживить вашу форму, включив в нее выпадающие списки и группы опций. Например, если нужно спросить пользователей, какой именно текстовый процессор они предпочитают, добавьте выпадающий список с перечнем приложений.

В первую очередь определите источник данных, связанный со списком. В базе данных **FormDB.odb** создайте новую таблицу, содержащую первичный ключ – поле ID и текстовое поле ListItem. Сохраните ее как **Table2**, откройте и сделайте в ней три записи, соответствующих трем текстовым процессорам. Следующий шаг – связать нашу форму с базой данных FormDB. Выберите Вид > Панели инструментов > Дизайн формы, и в появившейся панели Дизайн формы нажмите



› Мастер помогает добавлять в форму всяческие компоненты — например, выпадающие списки.

кнопку Навигатор форм. На палитре Навигатор форм перейдите к разделу Данные и в поле Источник выберите файл **FormDB.odb**. В выпадающем списке Тип содержимого источника укажите Таблица, а в списке Содержимое – Table2. Нажмите на кнопку Выпадающий список [Combo Box] на панели Элементы управления и нарисуйте на форме выпадающий список. Откроется мастер, где можно определить его параметры. Убедитесь, что у вас отмечено правильное поле (ListItems), и выберите опцию Нет, Сохранить значение только в форме. Нажмите Готово, и список готов к использованию. Извлечение значения из этого компонента просто и требует всего лишь две строки кода. Сначала надо отыскать список по его имени, а потом запросить значение:

```
ComboBox = Form.getByName("ComboBox")
ComboBoxValue = ComboBox.CurrentValue
```

Добавление группы кнопок-переключателей к форме так же просто, однако следует помнить, что:

1. Все переключатели в группе должны иметь одно и то же имя.
2. Для задания описания каждого переключателя следует использовать поле Метка в палитре Свойства.

Например, для выбора возможного ответа на вопрос «Как часто вы пользуетесь *OpenOffice.org*?» можно создать три кнопки-переключателя: «Ежедневно», «Раз в неделю» и «Раз в месяц».

Определение того, какая из кнопок активна в текущий момент, требует нескольких шагов. Во-первых, макрос должен найти все кнопки в указанной группе:

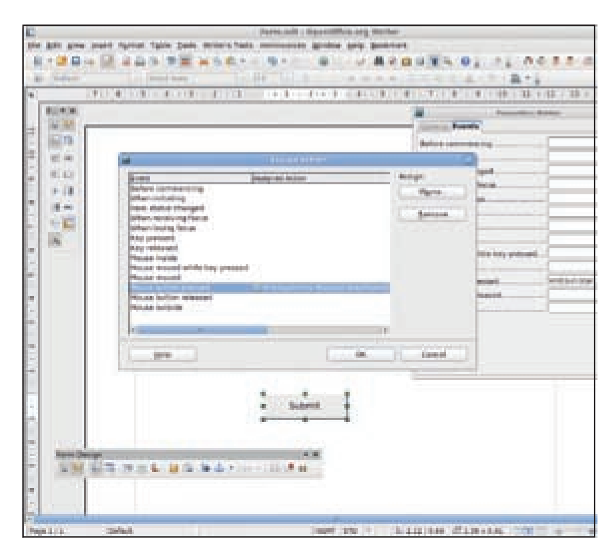
```
OptionGroup=Form.getGroupByName("Options", Opt())
```

Затем макрос должен в цикле **For ... Next** пройти через все пункты массива, от наименьшего (**LBound**) до наибольшего (**Ubound**), и определить состояние (**State**) каждой кнопки. Если свойство **State** возвращает True (это значит, что кнопка активна), макрос запрашивает ее значение **Label**:

```
For i=LBound(Opt) to UBound(Opt)
If Opt(i).state Then
OptionGroupValue=Opt(i).label
End If
Next
```

Обратите внимание, что все кнопки-переключатели в группе хранятся в массиве **Opt()**, который следует объявить в самом начале макроса с помощью оператора **Dim Opt()**. Вот как будет выглядеть полный макрос:

```
Sub WhichRadioButton
Dim Opt()
Doc=ThisComponent
Form=Doc.Drawpage.forms.getbyIndex(0)
OptionGroup=Form.getGroupByName("Options", Opt())
For i=LBound(Opt) to UBound(Opt)
If Opt(i).state Then
OptionGroupValue=Opt(i).label
End If
Next
MsgBox OptionGroupValue
End Sub
```



› **Работа с формой станет проще, если вы реализуете кнопку отправки данных, с помощью события Нажатие клавиши мыши.**

Проверка SQL-запросов

Прежде чем добавлять в ваш код какой-либо SQL-запрос, неплохо было бы посмотреть, работает ли он, в редакторе SQL. Для этого выберите в главном окне открытой базы данных Сервис > SQL, введите SQL-запрос в поле Команда [Command To Execute] и нажмите кнопку Execute. После этого вы увидите и результат, и объяснение возможных ошибок в поле Состояние [Status]. Эта возможность особенно полезна при отладке громоздких SQL-скриптов.

Настройка макросов

Созданные макросы вполне пригодны к применению, однако небольшая настройка способна облегчить их использование. Для извлечения данных из формы нужно запускать макрос вручную, через Сервис > Макросы > Выполнить макрос... и последующий выбор нужного макроса. Это не самый эффективный способ работы, но горю помочь легко: добавьте на форму кнопку Submit и свяжите ее с макросом. Для этого щелкните компонент Кнопка в панели Элемент управления и нарисуйте на форме кнопочку. В палитре Свойства переключитесь в раздел События и свяжите макрос с событием Нажатие клавиши мыши. Готово!

> «Небольшая настройка способна облегчить применение макросов.»

Теперь другая проблема: вы заметили, что форму надо сохранять перед тем, как макрос будет извлекать из нее данные? Если пользователь этого не сделает, макрос вернется с пустыми руками, или сохранит ранее введенные сведения. Чтобы предотвратить это, выполните проверку наличия в документе изменений с помощью свойства **isModified**. Если возвращаемый результат равен True, макрос сохраняет документ:

```
DocURL=Doc.getURL()
If Doc.isModified Then
Doc.storeAsURL(DocURL, Args)
End If
```

И, наконец, после извлечения данных можно очистить все поля формы. Это упростит следующему пользователю ввод сведений и предотвратит чтение того, что ранее вводил другой пользователь. Для этого напишем простой макрос и свяжем его с кнопкой очистки на форме:

```
Sub ResetForm
Doc = ThisComponent
Form = Doc.Drawpage.forms.getbyIndex(0)
Form.Reset()
End Sub
```

Вместо этого также можно добавить оператор **Form.Reset()** в конец макроса, который извлекает данные из формы. LXF

Python: Место

Ник Вейч вновь зачертил, но на сей раз идея хороша: найти самые фотогеничные участки вашей местности при помощи карт с геоданными…





Наш эксперт

Ник Вейч может заблудиться. Это научный факт. Но он всегда знает, где находятся его фотографии..

Верите вы или нет, но у меня есть подруга. Ей-Богу. Это реальный человек. Зовут ее Пэм, а живет она в Тампа, Флорида. В прошлом году она собиралась в Барселону и спросила, что там стоит фотографировать. А ведь это зависит от вкуса, верно? Кто-то хочет заснять обычные туристические объекты, как все, а иной предпочитает исследовать менее «затертые» части города. Вы можете прочесать сайты с фотографиями, но на самом деле вам нужна карта.

Множество фотоаппаратов и мобильников с камерами теперь внедряют в свои изображения геотэги – указывая широту и долготу – наряду с добавлением других Exif-данных. Сайты с фотографиями, подобные Flickr, записывают эту информацию и делают ее доступной. Так нельзя ли создать карту типа «где в этом городе люди делают снимки»?

WOE – это я

В 2008 году Yahoo открыла для разработчиков свою службу геолокации. Одной из ключевых концепций была иерархия уникальных ID для мест или областей этой планеты. Например, один ID идентифицирует такую вещь, как Эйфелева башня. Он будет связан с другими ID, таким как квартал, район, город, область, страна и т.д. Эти числовые ID получили название Where On Earth [Где на Земле] ID, или WOEID, и теперь широко используются в других геолокационных службах, включая Flickr. Более подробно о WOEID можно прочитать на http://tinyurl.com/yl7xz9g, но сейчас вам важно знать, что каждый город имеет WOEID, а многие еще и разбиты на районы, также имеющие свои WOEID.

Итак, как узнать WOEID какой-либо местности? Yahoo требует, чтобы вы зарегистрировались и получили ключ API, чтобы воспользоваться ее службами и найти ответ на данный вопрос. Однако, как мы говорили, Flickr также поддерживает WOEID, и, имея доступ к его «песочнице», мы сможем сделать тестовый запрос и получить WOEID интересующего нас города. Перейдите на http://tinyurl.com/ykbnmgb и введите 'Barcelona'. Из полученных данных станет ясно, что WOEID для Барселоны – 753692.

API Flickr также включают другой полезный метод с более длинным именем **places.getChildrenWithPhotosPublic()**. Он возвращает структурированный список данных, представляющий собой потомков той области, что вы указали. В случае города, потомками будут районы и/или, в некоторых случаях, почтовые индексы региона. Каждый из них снабжается долготой и широтой центра области и количеством общедоступных фотографий в ней.

Чтобы использовать для запросов API Flickr, необходимо иметь учетную запись на Flickr и обратиться за ключом API – см. врезку на стр. 79. Кроме того, раздобудьте модуль API Flickr для Python, что выливается просто в загрузку пакета для вашего дистрибутива или посещение http://stuvel.eu/projects/flickrapi.

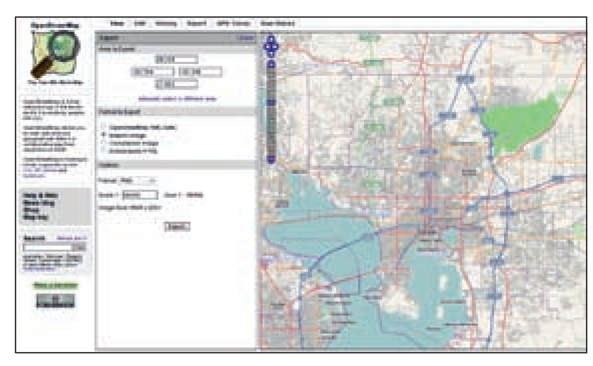
Заполучив эти компоненты, вы сможете увидеть, как работает ваш сборщик WOEID-изображений.

```
>>>import flickrapi
>>>apikey='bxxxxxxxxxx2'
>>>woeid=753692
>>>flickr = flickrapi.FlickrAPI(apikey)
>>>request=flickr.
places_getChildrenWithPhotosPublic(woe_id=woeid)
>>>list=request[0].getchildren()
>>>for item in list:
... if item.attrib['place_type_id'] == '22':
... print item.attrib['latitude'], item.attrib['longitude']
... print item.text
...
41.380 2.176
Ciutat Vella, Barcelona, Catalonia, ES, Spain
41.393 2.161
L'example, Barcelona, Catalonia, ES, Spain
41.383 2.183
Barceloneta, Barcelona, Catalonia, ES, Spain
41.386 2.177
Ribera, Barcelona, Catalonia, ES, Spain
41.406 2.179
```

» **Месяц назад** Мы внедрили Google в вашу файловую систему с помощью FUSE.

на карте





› **OpenStreetMap – лучший источник юридически чистых картографических сведений, которые можно отображать и публиковать.**

```
La Dreta De L'eixample, Barcelona, Catalonia, ES, Spain
41.379 2.167
...
```

Мы вызвали метод **getChildrenWithPublicPhotos**, который вернул результат. При тестировании его на сайте Flickr мы видели, что он возвращает объект с одним элементом, по имени **places**, содержащим список мест, соответствующих запросу. Чтобы получить сам список, раскроем объект при помощи метода **.getChildren** и проверим, пройдя по списку, что в полученных данных **place_type** означает соседство, тип 22. Причина тут в том, что некоторые области имеют перекрывающиеся объекты – например, области с почтовым кодом или районы, и некоторые снимки могут быть посчитаны дважды, исказив результат.

Метод **.attrib()** возвращает именованные значения (по ключу) из объектов отдельных мест и текст-описание, хранящийся в свойстве **.text**. Среди других значений – **photo_count** и WOEID района.

Если у нас имеются значения известных точек, можно отобразить их на диаграмме и показать наиболее фотографируемые области, не так ли? Теперь осталось приискать способ создания привлекательной графики из имеющихся данных.

Строим диаграмму

Если нужно построить контуры, гистограмму или график по точкам, каждый уважающий себя питонист первым делом обращается к модулю *Mathplotlib*. Как и его проприетарный кузен *Matlab*, *Mathplotlib* – это библиотека функций для преобразования данных в нечто приятное глазу. И опять-таки этот модуль Python обычно имеется в вашем дистрибутиве. Можно также установить модуль *Numpy*: он добавляет в Python несколько удобных возможностей и функций для визуализации данных, включая многомерные массивы.

Matplotlib содержит полезную функцию с именем **contourf**, которая просто берет список значений x, y и z и строит на их базе контурные цветные области вокруг точек с максимальным значением – по сути, это карта температур. Но прежде чем передать данные, следует устранить проблему: функция **contourf** работает только с данными в виде равномерной сетки, а наши точки и значения распределены случайным образом. Решение – создать пустую сетку данных, а затем наложить на нее наши значения при помощи какого-либо метода интерполяции. Вот почему модуль *Numpy* вам очень пригодится.

Метод **numpy.linspace()** возвращает список, заполненный набором числовых значений из заданного диапазона – а это нам и нужно для создания сетки x–y. Затем мы сможем воспользоваться функцией **griddata** из *Matplotlib* для интерполяции наших точек в нормальную сетку. Тогда наш код будет выглядеть так:

```
for item in list:
  if item.attrib['place_type_id'] == '22':
    y.append(float(item.attrib['latitude']))
    x.append(float(item.attrib['longitude']))
    z.append(int(item.attrib['photo_count']))
xmin=min(x)
xmax=max(x)
ymin=min(y)
ymax=max(y)
x += xmin, xmin, xmax, xmax
y += ymin, ymax, ymin, ymax
z += 0, 0, 0, 0
yrange=ymax-ymin
xrange=xmax-xmin
maxrange=max(xrange, yrange)
stepsize=maxrange/500
# создаем пустую сетку
xi = numpy.linspace(xmin,xmax,int(xrange/stepsize))
yi = numpy.linspace(ymin,ymax,int(yrange/stepsize))
# вносим данные в сетку
zi = griddata(x,y,z,xi,yi)
```

Здесь мы позаботились и о других двух моментах. Получение максимальных и минимальных значений для x и y дает нам границы прямоугольной области. Мы можем добавить эти точки (и нулевые значения) в наш массив данных, чтобы убедиться, что построенные контуры доходят до края карты. Кроме того, мы можем использовать эти данные для расчета шага нашей сетки. Не все города квадратные, поэтому мы изменяем размер шага в соответствии с занимаемой ими площадью. »

› **Flickr не только обладает превосходной документацией по API, но позволяет также делать пробные запросы с web-страницы.**

Осталось только воспользоваться функцией **contourf** для построения градиента вокруг областей и отобразить их (и сохранить в виде файла):

```
CS = plt.contour(xi,yi,zi,15,linewidths=0.5,colors='k')
CS = plt.contourf(xi,yi,zi,15,cmap=plt.cm.jet)
plt.colorbar() # рисуем цветную полоску
# отображаем точки данных
plt.scatter(x,y,marker='o',c='b',s=5)
plt.xlim(xmin,xmax)
plt.ylim(ymin,ymax)
plt.title('Barcelona')
plt.savefig('barcac.svg')
plt.show()
```

Здесь мы сперва построили контурную черную (**'k'**) линию толщиной 0,5, затем применили метод цветной заливки контуров. Параметр **cmap** – одна из предопределенных цветовых карт *Matplotlib*, дающая хороший диапазон от голубого через желтый до красного.

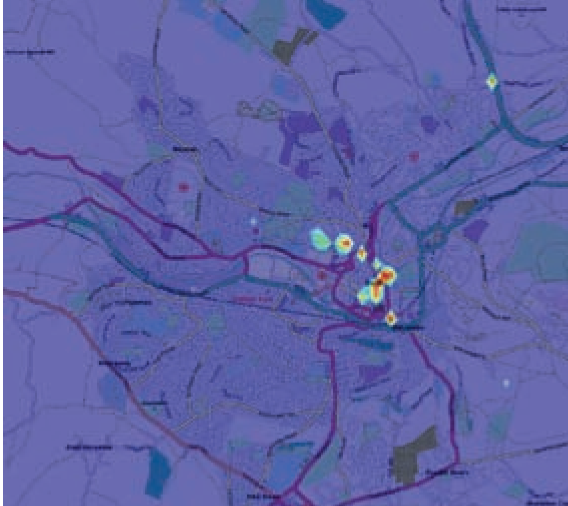
Остается наложить созданное изображение на карту. Можно бы зайти на Google Maps, но их карты, естественно, не ваша, а их собственность, поэтому лучше взять картографические данные от OpenStreetMap.

Простого метода изъятия PNG из указанного места нет, но вы можете загрузить его вручную, перейдя на вкладку Export на главной странице карты и введя широту и долготу местности, карту которой желаете загрузить. Выберите отрисовку OSM при наибольшем приближении.

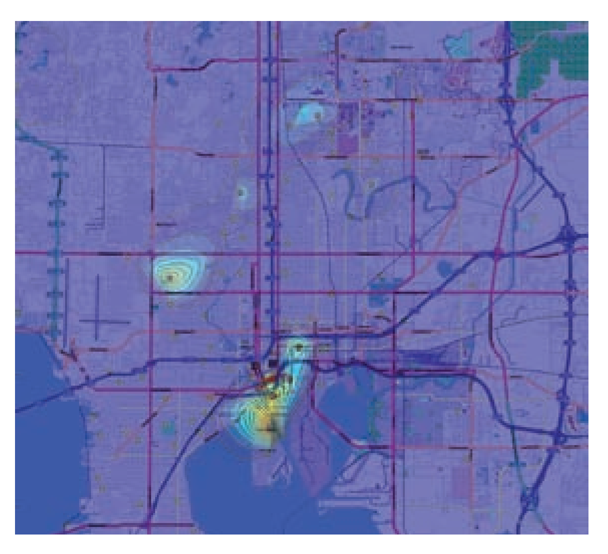
Изображение-карта и созданная карта температур могут быть совмещены в *GIMP*, *Scribus* или *Inkscape*. По-моему, в *Inkscape* это проще всего, потому что можно загрузить SVG-версию контурной карты, установить прозрачность 50% и растянуть ее над картой улиц. Соотношение сторон не будет таким же, поскольку в нашей контурной карте предполагается, что шаг по x и y одинаков, а на сферической поверхности Земли это не так, но для малых областей искажения минимальны. Тем не менее, чтобы получить информацию, соответствующую реальности, необходимо растянуть контур до размера карты.

Подождите, это еще не все

Этот подход работает только для большой местности, имеющей достаточно данных о соседях, чтобы в результате получить значимую информацию. Чтобы получить более точный вид или отрисовать области с малым числом соседей или вообще без них, необходимо действовать по-другому. Лобовой метод – загрузить информацию обо всех снимках, сделанных в требуемом районе. Для большого города это неподъемно, но, например,



› По этому рисунку с наложением точно видно, где люди делают снимки в Бате — хороший индикатор достопримечательностей.



› Хм. Навестить Тампа во Флориде, может, и интересно, но вряд ли я захочу там жить. Похоже, там маловато мест, достойных фотографии...

для городка Бат [курортное место, где находятся Башни **LXF**] вполне в пределах досягаемости.

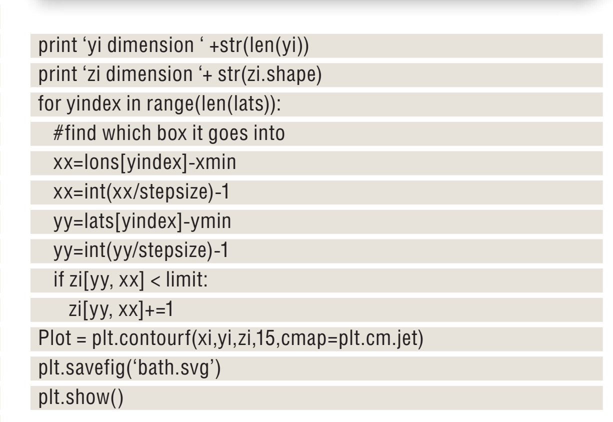
Основной метод Flickr **photos.search()** предоставляет механизм для извлечения данных только из некоторых WOEID. Затем информация об изображениях возвращается в виде постраничного списка, и по запросу в эти данные будут включены сведения о географическом положении.

Поскольку Flickr-сервера ворочаются со скрипом, лучше вытащить информацию для последующей обработки в файл – тогда нам не надо будет загружать данные о городе более одного раза. Их предварительный просмотр также даст нам возможность отсеять снимки с неверными геолокационными тэгами (иногда изображения помечаются как обладающие геоданными, но содержат нулевые широту и долготу) или с невысокой точностью.

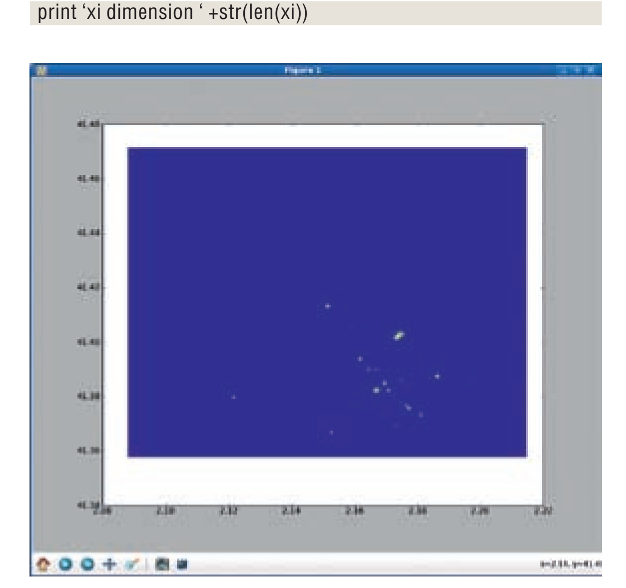
```
import flickrapi
apikey='bxxxxxxxx52' #your key here
out=open("bath.csv","w")
woeid=12056
geolist=[]
flickr = flickrapi.FlickrAPI(apikey)
#request=flickr.interestingness_getList(extras='geo',
per_page=500)
request = flickr.photos_search(woe_id=woeid, per_page=250)
foo=request.getchildren()
pages=int(foo[0].items()[3][1])
print pages
for i in range(1, pages+1):
  try:
    request = flickr.photos_search(woe_id=woeid, extras='geo',
per_page=250, page=i)
    foo=request.getchildren()
    bar=foo[0].getchildren()
    #bar теперь список фотоэлементов
    print 'attempt '+str(i)
    for pic in bar:
      if (pic.attrib['accuracy'])>14:
        lat=(float(pic.attrib['latitude']))
        lon=(float(pic.attrib['longitude']))
        if (lat, lon)!=(0.0, 0.0):
          out.writelines(str(lat)+', '+str(lon) +'\n')
    except:
      print 'skipped ' + str(i)
out.close
```

Этот скрипт пробегает все страницы результатов поиска данных и сохраняет широту и долготу в простом формате CSV. Мы можем воспользоваться этим файлом для создания более точной карты. Но нужно все-таки сгенерировать сетку данных; наш случай отличается отсутствием оси z. У нас просто есть точки, а значения в них одинаковы. Поэтому, воспользовавшись тем же подходом, что и раньше, мы получим плоское изображение. Для количественной оценки данных их следует сгруппировать. Мы можем сделать это, создав, как и ранее, пустую сетку, а затем пройти по всем точкам и определить, к какому «квадрату» новой сетки они относятся. Увеличение каждого отдельного значения массива z в случае принадлежности точки квадрату даст нам заполненность каждого из них, а на базе этого можно строить диаграмму.

```
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.cm as cm
####читаем данные
lats=[]
lons=[]
a = open('bath.csv', 'r')
for line in a.readlines():
  lats.append(float(line.split(',')[0]))
  lons.append(float(line.split(',')[1]))
print 'Data points: ', len(lats)
xmin=min(lons)
xmax=max(lons)
ymin=min(lats)
ymax=max(lats)
yrange=ymax-ymin
xrange=xmax-xmin
maxrange=max(xrange, yrange)
stepsize=maxrange/100
limit =int(len(lats)/20)
print 'area mapped:'
print xmin, xmax
print ymin, ymax
#define a new grid
xi = np.linspace(xmin,xmax,int(xrange/stepsize))
yi = np.linspace(ymin,ymax,int(yrange/stepsize))
zi = np.ones((len(yi), len(xi)))
print 'xi dimension ' +str(len(xi))
```



```
print 'yi dimension ' +str(len(yi))
print 'zi dimension '+ str(zi.shape)
for yindex in range(len(lats)):
  #find which box it goes into
  xx=lons[yindex]-xmin
  xx=int(xx/stepsize)-1
  yy=lats[yindex]-ymin
  yy=int(yy/stepsize)-1
  if zi[yy, xx] < limit:
    zi[yy, xx]+=1
Plot = plt.contourf(xi,yi,zi,15,cmap=plt.cm.jet)
plt.savefig('bath.svg')
plt.show()
```



› Это грубая сетка с данными для Барселоны, созданная скриптом. Чтобы придать ей смысл, просто наложите ее на карту.

Еще о чтении карт

Посмотрите Basemap (http://tinyurl.com/ygbyzfa), облегчающую отрисовку элементов в различных проекциях на реальной карте мира.

Документация Flickr API (www.flickr.com/services/api) не только информативна, но и предоставляет удобную функцию пробных вызовов API на web-странице.

О WOEID и других полезных сервисах читайте на Yahoo GeoPlanet (http://tinyurl.com/yfl7d4b).

Matplotlib, http://matplotlib.sourceforge.net, – это модуль Python для построения различных диаграмм.

И, наконец, посетите http://stuvel.eu/projects/flickrapi для лучшего интерфейса Python–Flickr.

Окончательная доработка

Есть еще кое-что, требующее доработки. Переменная **limit** была введена в связи с тем, что, скорее всего, немало изображений будут иметь одинаковые координаты. Даже в логарифмической шкале их количество «задавит» остальные части карты. В нашем скрипте **limit** вычисляется как процент от общего количества изображений в области – если результат не вдохновляет, поиграйте с этим значением, но 20 % вроде бы хорошо работает для большинства мест.

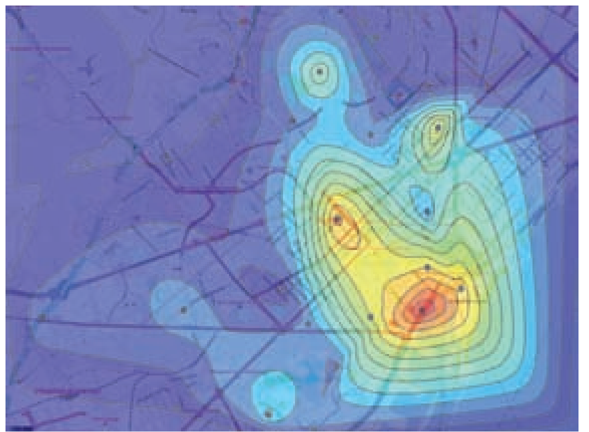
Еще один варьируемый параметр – количество квадратов нашей сетки. За него отвечает строка:

```
stepsize=maxrange/100
```

Фактически мы получаем 100 интервалов вдоль более длинной стороны. Другая сторона вычисляется в соответствии с этим значением. Вы можете счесть, что 100 – слишком мало. Увеличение шага сетки повышает гранулярность результата, но также и «шум» изображения, и в конце концов вы получите на экране практически двоичные данные. LXF

Скорая помощь

Команда *matplotlib* **plt.show()** открывает окно и останавливает выполнение скрипта, пока отображается картинка – вызывайте ее в самом конце вашего приложения.

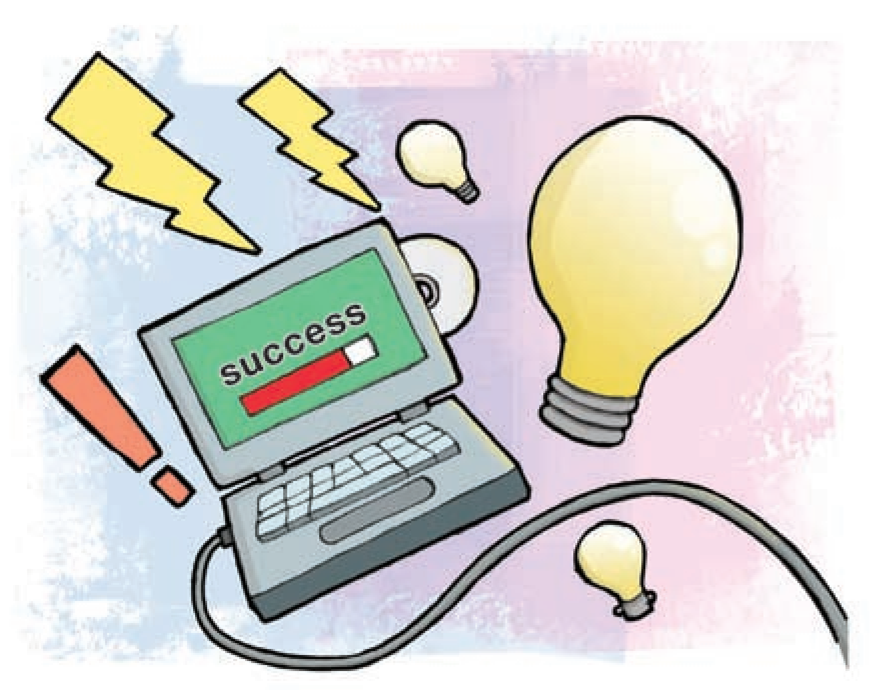


› Метод подсчета соседей дает гладкие контуры и хорошее представление о том, где люди делают снимки, но его трудно применить к конкретному месту.

» **Через месяц** Clutter – да-да, тот самый, на котором держится Moblin.

Phonon: И снова

GStreamer из **LXF125** показался вам недостаточно высокоуровневым? Не беда – **Андрей Боровский** предложит еще одну надстройку над ним (и не только) для тех, кто недолюбливает Gnome.





Наш эксперт

Андрей Боровский
И рад бы быть поближе к «железу», но наличие удобного каркаса все время возносит его на вершины абстракций.

В своих «Записках нечаянного революционера» Линус отмечал, что современные компьютеры хуже справляются с задачей наглядных обучающих пособий для хакеров, нежели те машины, на которых он сам начинал работу над Linux. В начале 90-х «систему» еще можно было окинуть одним взглядом. Любой достаточно квалифицированный программист не только мог, но и должен был перемещаться между уровнями оборудования и прикладного ПО. В наши дни такое вряд ли удастся кому-то на практике. Любой современный инструмент прикладного программиста – это несколько уровней обобщения и абстракции. Интересоваться, что скрывается там, под капотом, совершенно необязательно (а многообразие устройств и технологий обработки данных таково, что никто не сможет удерживать в голове сколько-нибудь значительную часть этой картины). Я иногда задумываюсь о том, что многие программисты двадцатилетней давности, с их интересом к «железу» и пониманием его работы, с их стремлением контролировать всю систему, возможно, не согласились бы заниматься прикладным программированием в наше время.

В нагромождении абстрактных слоев «виноваты» не только усложняющиеся технологии, но и пользователи. Сегодня они хотят, чтобы все источники мультимедиа открывались и контролировались одной программой, причем последняя должна предоставлять однородный интерфейс для управления такими разными мультимедиа-потоками, как аудиофайлы на диске, интернет-трансляции, простые видеоролики и данные DVD, разделенные на главы и имеющие по нескольку звуковых дорожек. В дополнение ко всему этому современная мультимедиа-программа должна быть легко расширяемой, то есть без проблем (и без перекомпиляции!) «брать на борт» новые кодеки и устройства. Можно сказать, что в то время, когда мультимедиа-технологии становятся сложнее, работающие с ними программы должны становиться проще.

Теория фононов

Система Phonon, которая продвигается проектом KDE и на данный момент является самым удобным мультимедиа-каркасом для *Qt*, представляет собой наглядный пример объединения нескольких слоев обобщений и абстракций. Основой Phonon (на платформе Linux) являются такие технологии, как ALSA и GStreamer (**LXF125**), которые сами представляют собой набор из нескольких уровней. Над этими технологиями Phonon надстраивает еще один уровень абстракции, оперирующий понятиями «поток данных» и «источник данных». В результате, от программиста, желающего написать с помощью Phonon, например, видеоплейер, требуется несколько строчек кода и не требуется понимания того, что при этом происходит с данными. Ситуация для меня лично грустная, но неизбежная.

В основе Phonon лежит идея, которая используется в большинстве современных мультимедиа-систем (например, DirectX): приложение представляет собой граф, в вершинах которого расположены компоненты обработки данных (источники данных, преобразователи, устройства ввода и вывода). Потоки данных передаются между компонентами по путям, определенным ребрами графа. Следует отметить, что интерфейсы Phonon еще не обрели окончательной стабильности. Даже документация *Qt* не всегда соответствует тому, что мы видим в библиотеке. Помимо поддержки Phonon, Qt Software разрабатывает собственную систему, QtMultimedia. В настоящее время ее работоспособность оставляет желать лучшего, и документация признает, что система Phonon более удобна для программистов.

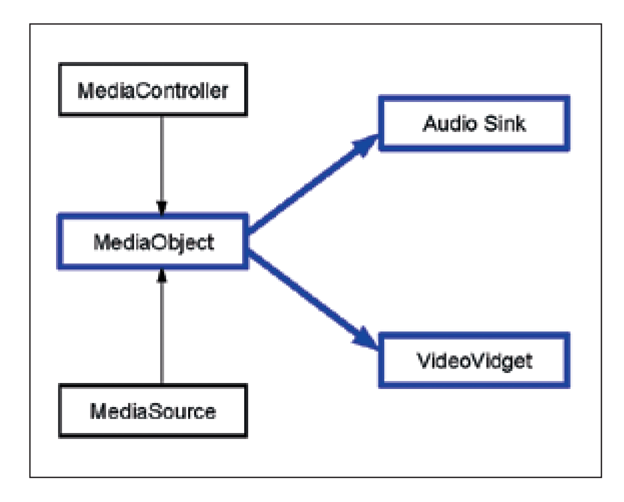


Рис. 1. Граф Phonon для видеоплейера.



видеоплейер

Чтобы не делать абстрактную модель еще более абстрактной, рассмотрим работу Phonon с точки зрения интерфейса подсистемы Qt-Phonon в *Qt 4.5–4.6*. Основными элементами графа Phonon являются объекты-источники данных (Phonon::MediaObject), компоненты-«стоки» [sinks], представляющие на уровне приложения устройства вывода данных (например, класс Phonon::VideoWidget, отображающий видео), и компоненты, выполняющие обработку данных, полученных из источника, перед передачей их компоненту-стоку (класс Phonon::Effect). Классы, предназначенные для создания узлов мультимедиа-графа, являются потомками Phonon::MediaNode.

Более подробно схема типичного мультимедиа-приложения Phonon (видеоплейер) показана на рис. 1. Компоненты, которые являются частью графа Phonon, обведены синей каймой. Кроме компонентов, входящих в граф, приложение использует вспомогательные. Объекты MeidaSource играют роль посредника между низкоуровневым источником данных и объектами MediaObject (интерфейс между интерфейсами!). Потомки MediaController позволяют выполнять сложное управление источником данных (например, навигацию по DVD).

Поскольку видеоплейер должен воспроизводить и видео, и звук, в нашем графе два стока. Если вы пишете программу, предназначенную для воспроизведения только звука, граф будет больше похож на цепочку компонентов.

Перейдем к практике

Чтобы программировать для Phonon и *Qt*, вам, помимо библиотеки *Qt*, понадобятся библиотека *libphonon* и файлы разработчика для нее, библиотеки и файлы разработчика ALSA (как минимум), а также компоненты мультимедийной системы, которые Phonon применяет для обработки данных. Любой разработчик серьезных мультимедиа-приложений для Linux сталкивается с одной специфической проблемой: в то время как в Linux есть программы и кодеки, позволяющие воспроизводить практически любой мультимедиа-контент, у кодеков наиболее популярных коммерческих форматов есть проблемы с лицензированием. Разработчики полностью открытых приложений могут просто игнорировать эти проблемы (практика показывает, что никто не станет их за то преследовать), но для разработчиков коммерческих продуктов (таких как *Qt*) подобный образ действий не подходит. Система Phonon решает проблему лицензирования просто: обработка мультимедиа-форматов перекладывается на другие компоненты системы, которые в терминологии Phonon именуются движками [backends]. Возможности Phonon в конкретной системе определяются возможностями этих движков (без корректно установленных движков система Phonon вообще работать не будет). В качестве движков Phonon может использовать, например, *GStreamer* и кодеки Xine. Документация по *Qt* упоминает только движки GStreamer и только открытые мультимедиа-форматы, намекая при этом, что возможно подключение и других форматов данных. На практике Phonon, при правильной настройке, поддерживает все форматы мультимедиа, которые понимают остальные мультимедиа-компоненты Linux.

Помимо самих компонентов *Xine* и *GStreamer* нужно установить библиотеки *phonon_gstreamer* и *phonon_xine*. Если у вас установлена новая версия KDE (4.2.2+) то, скорее всего, в вашей системе движки Phonon уже настроены.

Для подключения системы Qt-Phonon к проекту приложения *Qt* в файл ***.pro** необходимо добавить строку

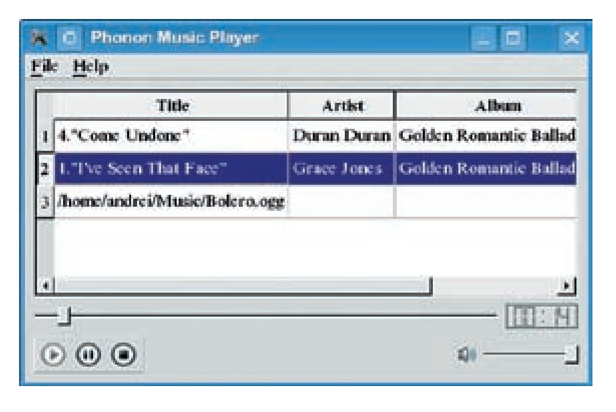
```
QT += phonon
```

Базовые классы Phonon определены в пространстве имен Phonon, которое становится доступным при включении в проект заголовочного файла <Phonon>.

Одно из важнейших преимуществ, которое мы получаем при использовании Phonon – кроссплатформенность. Система Phonon реализована для Windows (на основе DirectShow) и для Mac (на основе QuickTime). Как ни странно, заставить Phonon работать под Windows проще, чем под Linux: Qt SDK for Windows содержит все необходимые компоненты.

Привет, плейер!

Чтобы наделить приложение способностью воспроизводить аудиофайлы с помощью Phonon, в него достаточно добавить буквально три строчки кода. На диске вы найдете программу *SimpleAudioPlayer*, которая показывает, как это сделать. Все волшебство творит функция Phonon::createPlayer(), которая возвращает указатель на объект класса Phonon::MediaObject. Как мы знаем из вышеизложенного, его одного для воспроизведения звука недостаточно, но функция Phonon::createPlayer() создает вокруг данного объекта всю необходимую оснастку. Первый ее аргумент – категория устройства вывода, которое должен использовать плейер. Мы указываем Phonon::MusicCategory, то есть устройство, предназначенное для воспроизведения музыки. Вторым аргументом должна быть ссылка на объект Phonon::MediaSource, который поставляет объекту Phonon::MediaObject аудиоданные. »



› Рис. 2. Воспроизведение аудиофайлов с помощью Phonon.

Получив объект класса Phonon::MediaObject, мы вызываем его метод play() для воспроизведения аудио. Вместе все перечисленное выглядит так:

```
Phonon::MediaSource ms(fn);
m_player = Phonon::createPlayer(Phonon::MusicCategory, ms);
m_player->play();
```

В этом фрагменте переменная fn типа QString содержит имя аудиофайла, предназначенного для воспроизведения, а m_player представляет собой указатель на объект Phonon::MediaObject. Это почти все, что требуется для создания плейера. Осталась одна меленькая деталь. Поскольку Phonon использует для передачи данных между компонентами системы DBus, мы должны задать нашему приложению определенное имя с помощью метода setApplicationName() класса QApplication (что мы и делаем в функции main() нашей программы). Для остановки воспроизведения данных в нашем плейере можно воспользоваться методом stop() (а можно просто удалить объект-плейер).

Создание простейшего видеоплейера выглядит ненамного сложнее. Класс Phonon::VideoPlayer представляет собой виджет, который умеет воспроизводить видео. Его можно создать так:

```
videoPlayer = new Phonon::VideoPlayer(Phonon::VideoCategory,
parent);
```

Здесь parent – указатель на родительский виджет. Видео запускается на воспроизведение с помощью метода play() объекта Phonon::VideoPlayer. Для загрузки данных пригодится метод load(), аргументом которого является ссылка на объект Phonon::MediaSource, а можно вызвать перегруженный метод play() с тем же аргументом. Источник данных для воспроизведения аудио и видео может быть указан также с помощью объектов класса QUrl. Обратите внимание на то, что объект класса Phonon::VideoPlayer заботится о выводе не только изображения, но и звука.

Помимо начала и остановки воспроизведения, рассмотренные объекты-плейеры позволяют реализовать простую навигацию по источнику данных с помощью метода seek(). Но перед применением данного метода необходимо убедиться, что используемый источник данных допускает произвольный доступ. Для этой цели служит метод isSeekable().

Можно также реализовать простейшие списки воспроизведения. На уровне программы они представляются объектами QList; их элементы – объекты Phonon::MediaSource или QUrl (но не оба типа сразу, что не допускается самим синтаксисом C++).

› Рис. 3. Воспроизведение видео.

Если функциональность вашей программы ограничивается воспроизведением видео или музыки на устройстве, настроенном по умолчанию, на этом можно остановиться. В более сложных случаях нам придется самим строить мультимедиа-графы.

Построение графов

В состав Qt SDK входит программа *qmusicplayer*, которая представляет собой пример использования Phonon для создания аудиоплейера. В проигрывателе *qmusicplayer* мультимедиа-граф создается явным образом. Граф состоит из двух элементов: объекта-источника Phonon::MediaObject и объекта-стока Phonon::AudioOutput:

```
audioOutput = new Phonon::AudioOutput(Phonon::MusicCategory,
this);
mediaObject = new Phonon::MediaObject(this);
```

Для создания связи между двумя объектами используется функция Phonon::createPath():

```
Phonon::createPath(mediaObject, audioOutput);
```

У этой функции есть два аргумента-указателя на объекты Phonon::MediaNode: первый задает источник данных, второй – их приемник. После создания графа работа с ним выполняется так же, как и в рассмотренном выше случае. Помимо явного представления графа, в программе *qmusicplayer* есть еще один новый момент – использование объектов классов Phonon::SeekSlider и Phonon::VolumeSlider. Класс Phonon::SeekSlider представляет собой специальный виджет – ползунок, который одновременно управляет позицией воспроизведения в открытом файле и указывает текущую позицию (ни один современный мультимедиа-плейер не обходится без подобного элемента). Класс Phonon::VolumeSlider реализует, как вы, конечно, догадались, виджет-регулятор громкости. Объект Phonon::SeekSlider связывается с источником данных, поскольку именно он управляет навигацией в мультимедиа-файле. Для этой цели используется метод setMediaObject():

```
seekSlider = new Phonon::SeekSlider(this);
seekSlider->setMediaObject(mediaObject);
```

В свою очередь, Phonon::VolumeSlider связывается со стоком с помощью метода setAudioOutput():

```
volumeSlider = new Phonon::VolumeSlider(this);
volumeSlider->setAudioOutput(audioOutput);
```

Поскольку объекты классов Phonon::VolumeSlider и Phonon::SeekSlider являются виджетами, для них необходимо выполнить все те действия, который обычно производятся для настройки виджетов (то есть определить геометрию и указать параметры компоновки).

Простой граф программы *qmusicplayer* можно усложнить, добавив в него компонент-процессор. Последние, как мы знаем, преобразуют данные, добавляя к ним эффекты. Каждый компонент-процессор, добавленный в граф, отвечает только за один эффект. Для добавления компонента-процессора между источником данных и стоком нам понадобится объект, представляющий связь между компонентами графа. Его-то и возвращает функция Phonon::createPath():

```
Phonon::Path audioPath;
audioPath = Phonon::createPath(&mediaObject, &audioOutput);
```

Обратите внимание, что функция возвращает сам объект, а не ссылку и не указатель. Для добавления процессора мы используем метод insertEffect() класса Phonon::Path. Он существует в нескольких перегруженных вариантах, в каждом из которых, помимо прочего, передается объект-процессор класса Phonon::Effect. Метод insertEffect() можно вызвать несколько раз подряд, добавив, таким образом, несколько компонентов-процес-

«Каждый компонент-процессор отвечает только за один эффект.»

соров последовательно. Метод removeEffect() удаляет указанный процессор из графа.

Qt ничего не знает о том, какие именно эффекты доступны в вашей системе – с точки зрения библиотеки все они представляются одним и тем же объектом класса Phonon::Effect. Чтобы создать его, нужно получить описание требуемого эффекта. Перечень описаний всех доступных эффектов в виде списка QList <Phonon::EffectDescription> возвращает метод Phonon::BackendCapabilities::availableAudioEffects(). Каждый объект класса Phonon::EffectDescription содержит описание определенного эффекта. Метод name() этого класса возвращает описание эффекта на человеческом языке, и оно является единственным идентификатором, позволяющим определить тип аудиоэффекта. Иначе говоря, выбирать необходимый эффект должен пользователь,а не программа. Для этого можно создать виджет «раскрывающийся список» и заполнить его описаниями эффектов:

```
QList<Phonon::EffectDescription> availableEffects =
Phonon::BackendCapabilities::availableAudioEffects();
foreach(Phonon::EffectDescription desc, availableEffects) {
    ui->audioEffectsCombo->addItem(desc.name());
}
```

Теперь, если пользователь выберет элемент из списка, мы можем создать соответствующий объект-процессор и добавить его в граф:

```
int currentIndex = ui->audioEffectsCombo->currentIndex();
Phonon::EffectDescription chosenDesc =
availableEffects[currentIndex – 1];
Phonon::Effect * newEffect = new Phonon::Effect(chosenDesc);
audioPath.insertEffect(newEffect);
```

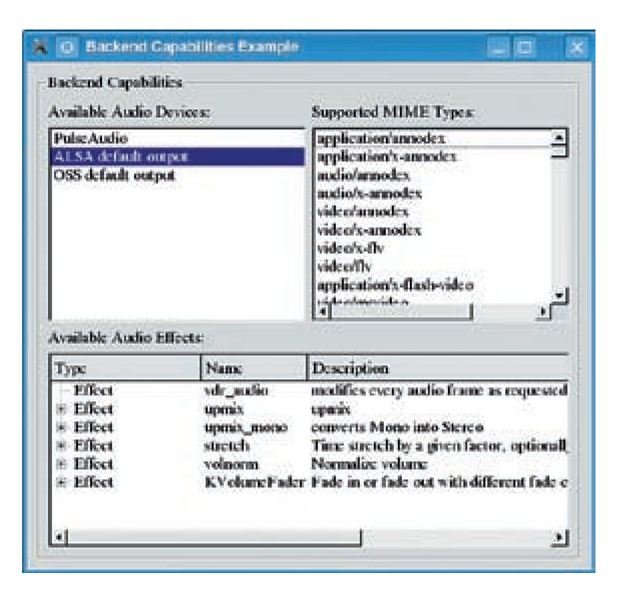
Хотя объекты-процессоры похожи на самостоятельные элементы графа, расположенные где-то между источником и стоком, на самом деле они реализованы в том же движке, с помощью которого компонент-источник обрабатывает данные. Таким образом, набор доступных процессоров зависит от настроек источника данных (в настоящее время компоненты-процессоры определены только для аудиоисточников). В общем, надо признать, что практической пользы от этих процессоров не очень много.

Аналогично строится граф для видеоплейера (пример *qmediaplayer* из Qt SDK). Хотя граф для видеоплейера должен включать в себя два стока, на практике его построение ненамного сложнее. Программа *qmediplayer* из набора примеров Qt SDK демонстрирует создание графа для одновременного воспроизведения видео и звука. Чтобы связать два компонента-стока с источником Phonon::MediaObject, мы дважды вызываем функцию Phonon::createPath(). Важно отметить, что при создании компонента-стока для вывода звука в видеоплейере указывается категория Phonon::VideoCategory. В созданном таким образом графе потоки видео- и аудиоданных синхронизированы.

Специфика работы с компонентом, предназначенным для вывода видео, заключается в том, что он должен быть встроен в пользовательский интерфейс.

Полезные дополнения

Программа *qmusicplayer* использует еще один объект класса Phonon::MediaObject (переменная metaInformationResolver). С помощью свойства metadata() этого объекта программа получает информацию о тэгах файлов, загруженных в список воспроизведения (для этой цели нельзя использовать объект mediaObject, так как получение подобных сведений не должно зависеть от состояния воспроизводимого файла). Метод metadata() возвращает контейнер, содержащий пары «ИМЯ–ЗНАЧЕНИЕ». Phonon использует одинаковые имена тэгов для всех аудиоформатов: они соответствуют определенным для Ogg Vorbis, так что при работе с другими аудиофайлами тэги будут приводиться в этот формат.



› Рис. 4. Перечисление устройств вывода звука.

Любая серьезная мультимедиа-программа должна предоставлять пользователю возможность выбора систем вывода данных. У Phonon есть своя система описания аудиоустройств, основанная на парадигме «модель–вид–контроллер». Для создания модели данных с описаниями аудиоустройств можно использовать и следующую тяжеловесную конструкцию:

```
Phonon::ObjectDescriptionModel<Phonon::AudioOutputDevice
Type> *model =new Phonon::ObjectDescriptionModel<Phonon::
AudioOutputDeviceType>();
```

В качестве параметра шаблона используется перечисление (enum) Phonon::ObjectDescriptionType. На данный момент оно может принимать два значения: Phonon::AudioOutputDeviceType – для описания устройств вывода звука и Phonon::EffectType – для описания эффектов. Получив объект-модель, мы связываем его с данными:

```
model->setModelData(Phonon::BackendCapabilities::
availableAudioOutputDevices());
```

Теперь модель и данные можно связать с объектом, отвечающим за представление, с помощью метода setModel() соответствующего объекта. Список объектов, представляющих сами устройства, можно получить с помощью той же функции, которая предоставляет данные для модели – Phonon::BackendCapabilities::availableAudioDevices(). Она возвращает список объектов Phonon::AudioOutputDevice, элементы которого содержат описания устройств вывода. Для установки выбранного устройства используется метод setOutputDevice() класса Phonon::AudioOutput. Как и в случае с эффектами, выбор устройства осуществляется пользователем на основе описания, составленного на естественном языке. На уровне программы Phonon мало что может сообщить нам о параметрах выбранного устройства.

Хотя современные средства разработки, с их высокоуровневыми функциями и наглядностью, возможно, и приводят к увеличению числа не очень толковых программистов, не следует думать, что толковых программистов от этого становится меньше. Возможность создать мультимедиа-плейер «одним щелчком мыши» появилась тогда, когда обычным мультимедиа-плейером уже нельзя было никого удивить. Программистам, желающим создать нечто новое, по-прежнему приходится быть оригинальными и изобретательными. LXF

» **Пропустили номер?** Узнайте на с. 107, как получить его прямо сейчас.

Privoxy: Рулим

Шашанк Шарма надел латексные перчатки таможенника королевской службы для вскрытия, досмотра и санации web-страниц фильтрами Privoxy.





Наш эксперт

Шашанк Шарма пишет о свободном ПО уже более четырех лет – для разных изданий, включая Linux.com. Он один из соавторов Beginning Fedora от Apress.

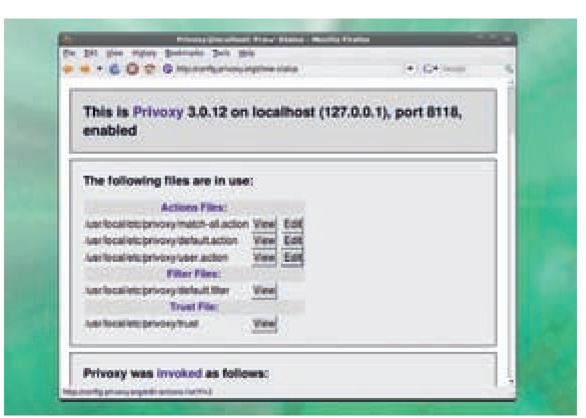
Нахождение сомнительного контента в Интернете не требует специальных знаний. Многие сайты, которые вы посещаете, содержат или ссылаются на вещи, неподходящие для показа в широком кругу. И мы говорим не только о сайтах-с-определенными-забавными-картинками: это может быть и нецензурный текст, надоедливые всплывающие окна или вредные сценарии JavaScript.

Privoxy может победить их вуду-магию: блокировать изображения и всплывающие окна, заменить на сайте текст, профильтровать содержимое, отключить JavaScript, а также многое другое, причем «на лету».

Прокси-сервер, который может выглядеть как сервером, так и приложением, выступает посредником между запросами клиента, просматривающего данные на других серверах. Он предоставляет две функции: скрывает клиента от остального мира и кэширует данные. *Privoxy* – это web-прокси, что намекает на его работу во Всемирной паутине.

Большинство дистрибутивов имеют пакет *Privoxy* в своих репозиториях, и вы легко установите его одной из следующих команд (в зависимости от вашей системы).

```
sudo apt-get install privoxy
su -c "yum install privoxy"
su - -c "urpmi privoxy"
```



› **Перезапускать Privoxy при каждом изменении конфигурационных файлов незачем. Изменения применяются «на лету».**

Если версия в репозитории окажется не самой свежей, вы также можете решить собрать *Privoxy* из исходных кодов. После установки демон *Privoxy* запускается через

```
/etc/init.d/privoxy start
```

Финальный шаг – оповестить о нем ваш браузер. Если у вас *Firefox*, выберите вкладку Дополнительные в меню Правка > Настройки. Затем, кликнув в Сеть на Настроить, зададим параметры подключения *Firefox* к Интернету. Используйте ручную настройку прокси-сервера и введите 127.0.0.1 для HTTP прокси и 8118 для Порта. Нажмите OK.

Теперь вы получаете доступ к интерфейсу *Privoxy* через http://config.privoxy.org или http://p.p. Там показаны текущие настройки и документация, и мы настоятельно рекомендуем прочесть ее для понимания возможностей *Privoxy*.

В директории **/etc/privoxy/** хранятся три конфигурационных файла, определяющих, как модифицировать содержимое просматриваемых страниц. Основной из них называется **config**. В нем необходимо изменить всего пару строк, а именно **enforce-blocks** и **enable-edit-actions**.

Privoxy включает контроль за web-страницами в двух видах: действия и фильтры. Фильтры могут перезаписать контент страницы, HTML или любой включенный в нее JavaScript. Правила определяют действия, выполняемые над cookies, блокируемыми баннерами, изображениями, всплывающими окнами и другими объектами, заданными в файле действий. Файл **default.action** содержит вышеуказанные правила, но изменять его не рекомендуется. Вы можете управлять только **user.action**, который имеет преимущество над остальными файлами действий, так как загружается последним.

Вы можете включать действия и выключать их, используя знаки + или - соответственно. Например, **+block** будет блокировать сайты, а **-block** – нет.

» **Месяц назад** С помощью Perl мы переделали ваш компьютер в ТВ-тюнер.

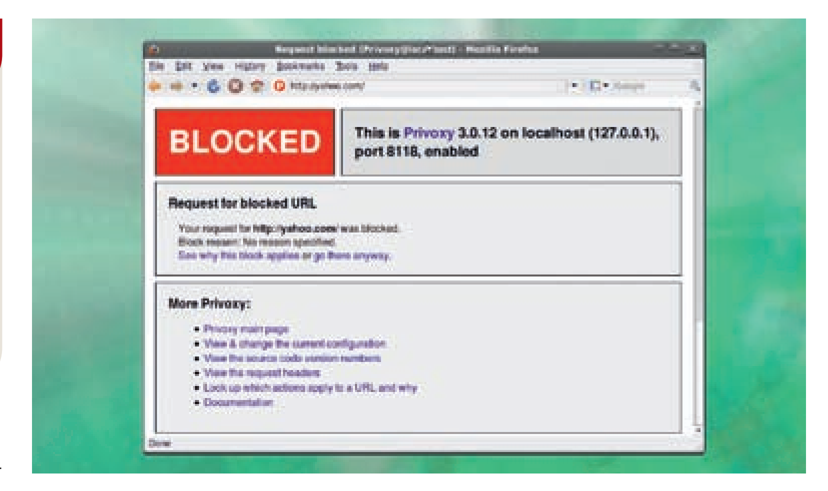
web-доступом

Редактируем в браузере

Privoxy позволяет редактировать свои файлы конфигурации через web-интерфейс в браузере. Эта функция отключена по умолчанию, но может быть активирована путем правки файла **config**. Пролистайте вниз до раздела 4.5 файла **config**, озаглавленного **enable-edit-actions**, и измените в строке **enable-edit-actions 0** на **1**, не забыв сохранить изменения.

Теперь, вновь зайдя на страницу http://config.privoxy.org/show-status, вы заметите кнопку Правка [Edit] рядом с текущим используемым файлом.

Разумеется, пользователям локальной сети незачем разрешать редактирование файлов конфигурации.



› Изменение значения enforce-blocks с 0 на 1 в файле config выключит этот параметр в любом случае.

Ограничение доступа

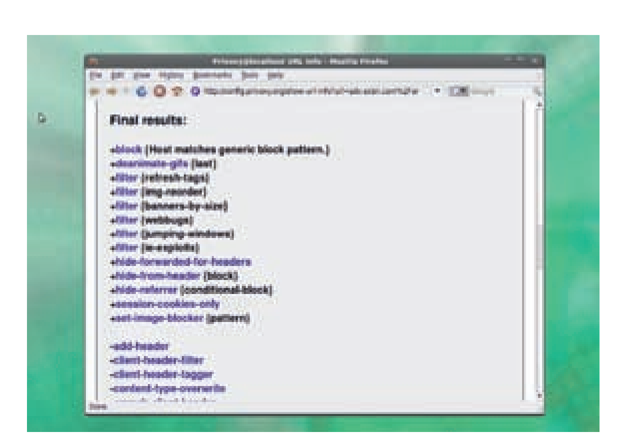
При первом запуске *Privoxy*, без правок конфигурационных файлов, вы можете заметить, что загрузить ничего не получается. Вместо этого вам выдается сообщение: «Request blocked by Privoxy: Host matches generic block pattern. See why this block applies or go to the link anyway».

При щелчке на ссылке Why This Block Applies *Privoxy* отобразит настройки, которые используются для блокировки соответствующего содержимого. Каждый элемент в списке имеет ссылку на пояснение в руководстве пользователя, помогающее понять, зачем используется то или иное правило и как оно применяется к текущей странице.

К сожалению, самое распространенное применение *Privoxy* – это блокировка изображений и другого содержимого по его размеру. Например, в конфигурации по умолчанию Slashdot, Digg и другие сайты отображаются отлично от оригинала, иногда теряя рекламу от Google.

Так как не каждому понравится, что в сети завелся некто, пожирающий трафик, например, на просмотр роликов YouTube, то при работе с *Privoxy* вы можете создать список доверенных сайтов, и доступ будет разрешен только к ним.

Самый простой, быстрый и по-настоящему действенный способ ограничить доступ – это добавить в файл **user.action** такой код:



› При конфликте правил, относящихся к одному и тому же содержимому — например, сразу +block и -block — применяется последнее.

```
{ +block }
/ # Так вы заблокируете *все* сайты
```

Эти команды блокируют доступ ко всем сайтам; но если с возрастом вы смягчились и согласны отрезать только некоторые, перечислите их после строки **{ +block }**.

```
{ +block {Нечего тут глазеть. Работать надо} }
scorecard.example.com
```

Зайдя на scorecard.example.com, пользователь, кроме сообщения о блокировке, получит уведомление о ее причине. (Вы можете сделать его посмешнее, чем здесь – мы теориям комических сообщений об ошибках не обучались.)

Если вы предпочитаете использовать «белый» список вместо «черного», ваш код будет выглядеть так:

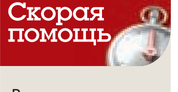
```
{ -block }
lxer.com
slashdot.org
```

Однако, несмотря на наличие «белого» списка, при использовании этого метода вы будете сильно ограничены в доступе к сайтам, потому что сможете попасть на slashdot.org, но не в его поддомены, такие как developers.slashdot.org.

Альтернативным вариантом будет использование механизма файла **trust**, в котором перечисляются все доверенные сайты. При занесении URL в файл **trust** необходимо предварить его знаком + или ~. Тильда (~) ограничивает доступ к сайту и его подпутям. Любые внешние ссылки на сайте или страницах будут запрещены. Скажем, **~lxer.com** даст вам неограниченный доступ внутри lxer.com, но не позволит следовать по новостным ссылкам на внешние сайты.

В свою очередь, + описывает сайт как заслуживающий доверия. Преимущество здесь в том, что на недоверенные сайты – а это все сайты, не перечисленные в файле **trust** – можно попасть только с ссылающегося на них сайта со знаком +. Так, **+lxer.com** означает, что вы можете проследовать по всем внешним ссылкам, и они будут автоматически добавлены в список доверенных.

Например, чтобы позволить доступ к example.com и чтобы «белый» список был заполнен доменами по ссылкам с example.com, используйте такую строку:



В зависимости от установки *Privoxy*, конфигурационные файлы могут храниться в директории **/etc/privoxy** или в **/usr/local/etc/privoxy/**.

»

```
# Пример: для разрешения доступа к example.com и появления белого списка доменов
# на основе ссылок с example.com, расскомментируйте эту строку:
# +example.com
+lxer.com
# Следующие две строки дают пользователям доступ
# к CGI страницам Privoxy, без автоматического доверения ссылок
~config.privoxy.org
~p.p
~tags.expo9.exponential.com # По ссылке с: http://lxer.com/
~www.unixmen.com # По ссылке с: http://lxer.com/module/newswire/view/124423/index.html
```

Заметьте, что новые ссылки в конце файла начинаются со знака **~**, то есть они не являются доверенными. Обычно ваш файл **trust** растет со временем, в зависимости от того, насколько либерально вы расставляете знак **+**.

Скорая помощь

Шаблоны подстановки – основа конфигурационных файлов, поэтому не удивляйтесь, увидев множество точек, звездочек и квадратных скобок.

Блокируем cookies и баннеры

Если вы параноидально относитесь к безопасности, или просто хотите обеспечить блокировку всех cookies со всех просматриваемых сайтов, воспользуйтесь **+crunch-incoming-cookies** для их отключения. Это удалит заголовок **Set-Cookie: HTTP** из всех запросов сервера. **+crunch-outgoing-cookies** работает похоже, отключая возможность сайтов читать любые HTTP-cookies с вашей машины и удаляя все заголовки **Cookie: HTTP** из клиентских запросов.

Эти правила заданы по умолчанию в файле **default.action**. Чтобы создать из них исключение – например, разрешить cookies от некоторых сайтов, блокируя остальные – следует дописать в **user.action** нечто вроде.

```
-crunch-incoming-cookies
.linuxquestions.org
.en.wikipedia.org
```

Есть несколько способов записи URL-адресов для web-сайтов внутри файла действия. Так, точка в начале будет означать применение правила ко всем поддоменам, которые заканчиваются на **.url.com**. При задании URL допускаются также символы подстановки. Например ***ing.com** будет совпадать с bling.com, ring.com и т. д.

Избавление от баннеров, похоже, является основной целью *Privoxy*, но при этом, без правки конфигурации по умолчанию, содержимое некоторых сайтов не будет видно вообще – из-за правила **all-encompassing** в файле **default.actions**. Поэтому для таких сайтов нужно будет добавить исключения в файл **user.actions**.

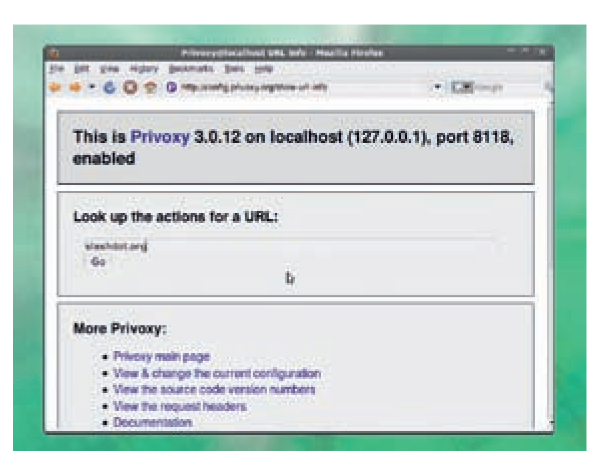
Используем псевдонимы

Мы упоминали, что файл **default.actions** редактировать нельзя и что все исключения и новые правила должны быть добавлены в файл **user.action**. Теперь приведем причину такого поведения.

Псевдоним – это любое действие, который вы создаете комбинированием двух или более существующих. Так, при желании написать исключение для нескольких действий над какими-либо сайтами вы можете ввести каждое из них по отдельности, или же создать один псевдоним для всех данных действий и затем использовать его, сэкономив на наборе текста. Например, **cookiesads = -crunch-all-cookies -session-cookies-only -filter{content-cookies} -filter{banners-by-size} -filter{banners-by-link}** – это псевдоним для нескольких различных действий. Каждый раз, когда нужно описать каждое из этих действий для сайта, вы можете использовать псевдоним в файле **user.action**:

```
{cookieads}
.linuxquestions.org
.digg.com
.slashdot.org
```

Вам не нужно использовать псевдонимы со знаками + или -, так как они превращаются в реальные действия подстановкой. Вы можете назвать псевдоним как угодно, без ограничений на длину, но не используя пробелы, табуляцию, фигурные скобки или знак =. Псевдонимы уникальны для каждого файла **action** и задаются в его начале.



› Действия, применяемые к данному URL, выясняются щелчком на Look up which actions apply to a URL and why [Какие действия были применены к URLи почему].

Для блокировки баннеров используются две основные техники – banners-by-size [баннеры-по-размеру] и banners-by-link [баннеры-по-ссылкам]. Оба метода заданы в файле **default.filter**.

По умолчанию *Privoxy* удаляет баннеры основных размеров, а именно, вертикальные 160×600, 250×250, 300×250, 468×60, и многие другие. Это покрывает почти всю известную баннерную рекламу, но иногда приводит к ошибочному отключению не рекламных изображений. Если это произошло, или вы не хотите блокировать рекламу на определенных сайтах, можете написать исключение.

Соответствующий фильтр в файле **action** имеет вид **+filter{имя}**. То есть, если вы не хотите применять фильтр к каким-либо сайтам, создайте новый раздел **-filter{banner-by-size}** в файле **user.action** и перечислите сайты ниже.

Поскольку основным (читай: единственным) источником дохода в Интернете является реклама, мы полагаем, что блокировать объявления исключительно на основе их размера негуманно. Фильтра banners-by-link должно быть достаточно для большинства пользователей, если вы хотите удержать популяцию territory кликозависящих сайтов в рамках.

Заменяем «на лету»

Когда дело касается web-страниц, файл **default.filters** начинает вести себя как Судья Дредд: судит, приговаривает и приводит в исполнение. Он не слушает доводов. Судейство – не повод для дискуссии и не может быть обжаловано.

Многие из фильтров, описанных в **default.filter**, легки в интерпретации и понимании. Так, **jumping-windows** защищает окна от изменения размера. Фильтр **all-popups** блокирует всплывающие окна HTML и JavaScript. Это, однако, следует использовать с осторожностью, так как многие сайты используют JavaScript для открытия новых полезных окон. Бывали на сайте знакомств? Вы не сможете посмотреть профиль человека и взглянуть на его физиономию, если отключите всплывающие окна.

Что выделяет этот и другие основные фильтры – это оператор Perl **s///**. Вы можете создать фильтр и использовать синтаксис оператора **s///**, заменяя любые части в web-странице, включая JavaScript:

```
FILTER: js-events Убивает все JS события и таймеры
(Очень разрушительно! Только для экстра-грязных сайтов).
s/(on(event\.)((mouse(over|out|down|up|move))|(un)?load|contextmenu|selectstart)/never/ig
# Не события, но тоже могут применяться во вред:
s/(alert|confirm)\s*\(/concat(/ig
s/settimeout\(/concat(/ig
# Убить всплывающие окна OnUnload. Ура!
```

```
s/(<body\s+[^>]*)onunload/$1never/siU
sl(<script.*)window\.onunload(?=.*</script>)l$1neverlsigU
```

Вы можете использовать похожий синтаксис для замены каждого вхождения слова ‘просто’ на ‘жуть’ с помощью **s/просто/жуть/g**.

Обходим цензуру

Одна из популярных схем заключается в использовании *Privoxy* совместно с *Tor*. Вы можете рассматривать *Tor* (The Onion Router) как телефонную будку для супергероев: он дает каждому новое лицо. Вы входите в *Tor* с вашим собственным IP-адресом, а выходите с другим. Иначе говоря, вас нельзя отследить.

Сеть *Tor* состоит из ретрансляторов, запущенных добровольцами по всему миру. Вы получаете к ней доступ с помощью специального прокси (также называемого *Tor*) на вашей машине. Идея заключается в передаче данных через различные узлы сети, прежде чем они достигнут назначения. Для защиты данных и их источника *Tor* применяет шифрование на каждом из узлов. С точки зрения пункта назначения трафик идет из последнего узла.

Причина популярности связки *Tor* и *Privoxy* лежит в возможности обхода цензуры. Если сайт блокирует вашего провайдера, или ваш провайдер блокирует какие-то сайты, эта схема может помочь вам получить доступ в обход блокировок.

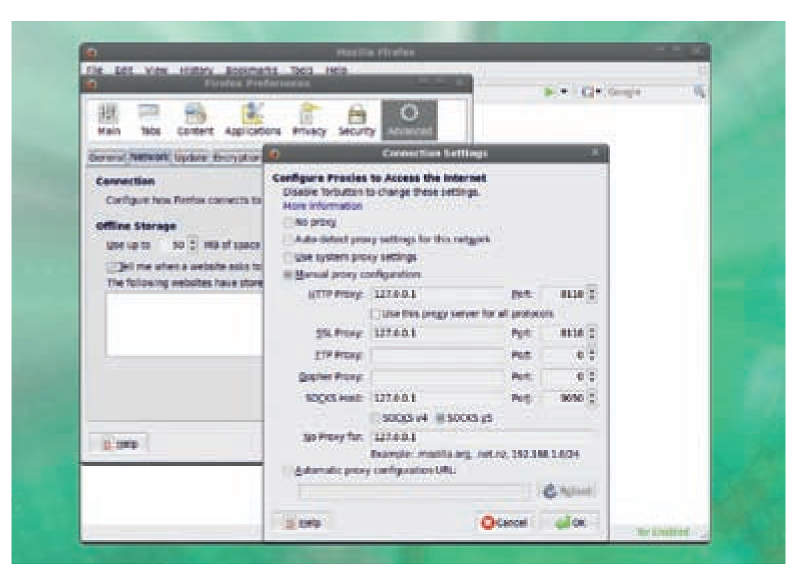
Вы можете установить *Tor* в вашу систему одной из следующих команд, в зависимости от вашего дистрибутива:

```
su -c “yum install tor”
sudo apt-get install tor
su - -c “urpmi tor”
```

Вам, возможно, понадобится *Torbutton*, расширение *Firefox* для включения и выключения *Tor*, а также *Tork* – графический инструмент настройки *Tor*.

Расширение *Torbutton* добавит кнопку в нижний правый угол окна *Firefox* в строку состояния. Следующим шагом будет настройка *Privoxy* для *Tor*, поэтому откройте файл **config** в вашем любимом текстовом редакторе и пролистайте его до Раздела 5.2:

```
#    Для соединения Privoxy и Tor, если оба запущены на одной машине,
#    вы должны использовать что-то типа этого:
#    forward-socks4a   /           127.0.0.1:9050 .
```



› *Firefox* **в связке с** *Tor* **будет брать настройки для SSL-прокси и поля SOCKS Host из конфигурационного файла.**

Останется только раскомментировать последнюю строку. Так как *Tor* использует публичную сеть Tor для анонимизации вашего трафика, вы не можете применять его для доступа к вашей локальной сети. Поэтому, если доступ к ней вам все-таки нужен, раскомментируйте в файле **config** *Privoxy* такие строки:

```
#     forward      192.168.*.*/     .
#     forward      10.*.*.*/        .
#     forward      127.*.*.*/       .
```

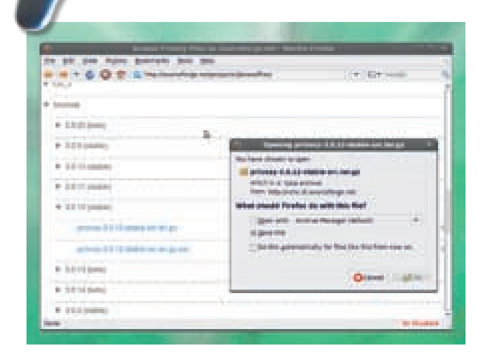
Большинство браузеров теперь предоставляют многое из того, что умеет *Privoxy*: например, блокировку всплывающих окон и отключение изменения их размера, а кое-что получают с помощью расширений. Количество времени, усилий и терпения, необходимых для нормальной настройки *Privoxy*, на этом фоне может показаться вам пустой тратой времени.

Но *Privoxy* позволяет гораздо больше в части контроля web-страниц. Если принять во внимание все его параметры и тот факт, что многие «умолчания» ссылаются на отличную документацию, в итоге вы получите весьма впечатляющий и полезный инструмент. LXF



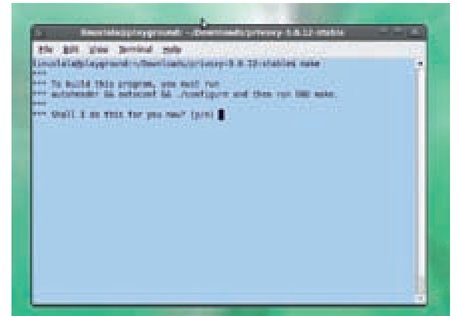
Для чтения файла сообщений используйте *Vi*: он подсвечивает сообщение, и читать его проще.

Шаг за шагом: Установка Privoxy из исходных кодов



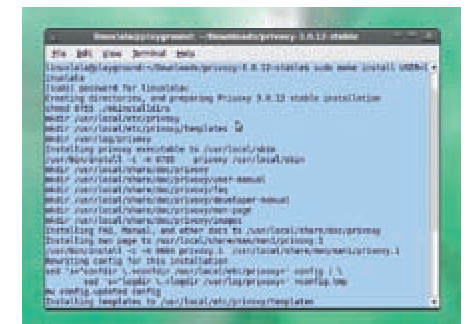
1 Загрузите архив

Некоторые дистрибутивы до сих пор содержат в своих репозиториях бета-версию *Privoxy*, и вы не найдете текущий стабильный релиз в Jaunty. Поэтому скачайте архив исходных текстов **3.0.12.tar.gz** с сайта проекта.



2 Сделайте меня

Перед **make install** вы должны запустить **autoheader**, **autoconf**, **./configure** и **make**. Запустите **make** сразу, и вас попросят выполнить остальные. Если вам не нужно передавать в **./configure** никакие параметры, наберите **y** и нажмите Enter.



3 Задайте пользователя

Privoxy предлагает создать отдельного пользователя и группу *Privoxy* и использовать их на этапе установки. Иначе, наберите **make install USER=linuxlala**, чтобы настроить *Privoxy* для указанного пользователя.

» **Через месяц** Повитаем в облаках вместе с Amazon EC2.



Ответы

Есть вопрос по Open Source? Пишите нам по адресу answers@linuxformat.ru

1 Незваный гость

В У меня установлена gOS как гостевая система в VirtualBox, на хосте Kubuntu 9.04, и я наткнулся на проблемы при установке гостевых расширений Guest Additions. Я запустил sudo apt-get install dkms в gOS и смонтировал образ Guest Additions там же. Но когда я захожу в Guest Additions и запускаю sudo sh ./VBoxLinuxAdditions-x86.run, то получаю следующее:

```
VirtualBox 2.1.4 Guest Additions installation
Please install the GNU compiler.
Problems were found which would prevent the
Guest Additions from installing.
Please correct these problems and try again.
```

Мэтт Бартон [Matt Burton]

О В пакет Guest Additions включены некоторые модули драйверов, версии которых должны быть согласованы с версией установленного у Вас ядра. Если в пакете нет подходящих предварительно собранных модулей, то они будут скомпилированы из исходных кодов. Для новичка это может звучать устрашающе, хотя процедура очень простая и целиком контролируется установочным скриптом. Но справиться с ситуацией, когда отсутствует одна из утилит для сборки модулей, скрипт не может. В таком случае он сообщает об отсутствии компилятора GNU, но не говорит, как это исправить. Решение – установить пакет **build-essential**. В Debian и его производных он отвечает за установку разных инструментов, предназначенных для компиляции ПО. **Build-essential** отсутствует в gOS в меню Установка/удаление программ [Add/Remove Software], но его можно добавить с помощью *Synaptic* через меню Системные настройки > Администрирование [System Preferences > Administration], или же через консоль:

```
sudo apt-get install build-essential
```

После его установки скрипт *VBoxLinuxAdditions* должен сработать правильно. НБ



› Для установки Guest Additions в *VirtualBox* в гостевой ОС необходим набор инструментов компиляции.

2 Обновление в сетинет

В Спасибо за отличный совет про сценарий загрузки пакетов из Synaptic для компьютера без сетевого подключения из номера LXF123. Я запустил скрипты и, чтобы загрузить пакеты, перенес их на борту моего допотопного ноутбука в нашу локальную беспроводную сеть. Все сработало действительно прекрасно, причем удобнее, чем с APTonCD. А вопрос у меня такой: как обновить список репозиториев на компьютере, отключенном от сети?

Metta Forest Monastery

О Такое возможно, но тут придется прибегнуть к магии командной строки для создания сценария загрузки, работающего под Windows. К счастью, у нас в Linux есть богатый выбор утилит для обработки текстовых данных, преобразующих вывод одной программы в формат, пригодный для работы с другой. *Apt-get* – механизм, стоящий за *Synaptic* – имеет опцию для вывода списка файлов, которые нужно загрузить:

```
apt-get --print-uris update
```

Опция **update** обычно обновляет базу данных пакетов на Вашем компьютере, но по флагу **print-uris** просто выводится список файлов, которые надо скачать, в виде нескольких строк наподобие такой:

```
'http://security.ubuntu.com/ubuntu/dists/ jaunty-security/main/binary-i386/Packages. Bz2' security.ubuntu.com_ubuntu_dists_jaunty-security_main_binary-i386_Packages 0 :
```

Первый компонент строки – URL-адрес, второй – имена, под которыми будут сохранены файлы. Одни из этих файлов текстовые, они остаются неизменными, а другие – архивы bzip2, предназначенные к распаковке. Для извлечения информации и превращения ее в скрипт закачки мы прибегнем к команде **awk**.

```
apt-get --print-uris update | sed "s/'//g" | awk
'!/bz2/ {print "wget " $1 " --output-document "
$2 "\r"}' >update.bat
```

Наши эксперты

» Мы найдем ответы на любой вопрос — от проблем с установкой системы или модемом до сетевого администрирования; главное — спросить!



Нейл Ботвик
Владелец ISP и экс-редактор дисков для нашего журнала, Нейл считает, что в Linux он от скуки на все руки.



Майк Сондерс
Майк был одним из создателей прототипа LXF – Linux Answers. Его специальности – программирование, оконные менеджеры, скрипты инициализации и SNES.



Пол Хадсон
Пол – местный супер-программист, и он может и хочет управиться со всеми вашими проблемами по части web и баз данных.



Грэм Моррисон
Когда он не обозревает кучи программного обеспечения и не халтурит с MythTV, Грэм готов дать ответ касательно любого оборудования и проблем виртуализации.



Валентин Синицын
В редкие свободные минуты главный редактор нашего журнала обычно запускает *mcedit*, чтобы отшлифовать какое-нибудь открытое приложение. Его любимая тема – настольный Linux.



Юлия Дронова
Если компьютер у Юлии не занят выполнением команды *emerge*, она спешит применить его для модерирования ЛинуксФорума.

КУДА ПОСЫЛАТЬ ВОПРОСЫ

Пишите нам по адресу: answers@linuxformat.ru **или спрашивайте на форуме:** www.linuxformat.ru

```
apt-get --print-uris update | sed "s/'//g" | awk '/
bz2/ {print "wget " $1 " --output-document "
$2 ".bz2\r"}' >>update.bat
```

Первая часть команды – шаблон, которому нужно искать соответствия, а если впереди стоит знак !, то несоответствия. Поэтому первая команда выбирает все строки, не содержащие **bz2**, и генерирует вызов **wget**, которая загружает первый компонент вывода **apt-get** и сохраняет его под именем, указанном во втором компоненте вывода. Вторая команда делает то же самое для всех строк, содержащих **bz2**, и добавляет к имени файла подходящее расширение. В обоих случаях в конце добавляется символ **CR**, поскольку этот скрипт будет использоваться на Windows-машине, а она ожидает, что строки разделяются **CRLF**, а не просто **LF**, используемым в Linux. Заметьте, что команды перенаправления отличаются: первая строка создает новый файл с оператором >, а вторая – с >>. Часть команды, за которую отвечает **sed**, удаляет одиночные кавычки, обрамляющие адреса URL, поскольку в Windows они создадут проблему. Этот шаг можно пропустить, как и добавление **CR** в конце каждой строки, если для загрузки Вы будете использовать машину с Linux.

Скопируйте файл **update.bat** на USB-флэшку и поместите туда же программу *wget.exe* для Windows: ее можно загрузить с сайта http://users.ugent.be/~bpuype/wget. Подключите флэшку к компьютеру с Windows и двойным щелчком запустите **update.bat**, чтобы закачать файлы. Смонтировав после этого флэшку в Linux, распакуйте файлы с помощью команды:

```
bunzip2 *.bz2
```

и затем скопируйте все загруженное в **/var/lib/apt/lists**. В заключение обновите базу данных пакетов с помощью:

```
apt-get --no-download update
```

Если Вы скачаете списки репозиториев в интернет-кафе (или еще где-нибудь) за тот же сеанс, что и пакеты обновлений, установите последние до обновления репозиториев. В противном случае может оказаться, что скачанные Вами пакеты не являются самыми свежими, и труды по их загрузке пошли насмарку. **НБ**

3 Кодирование DVD

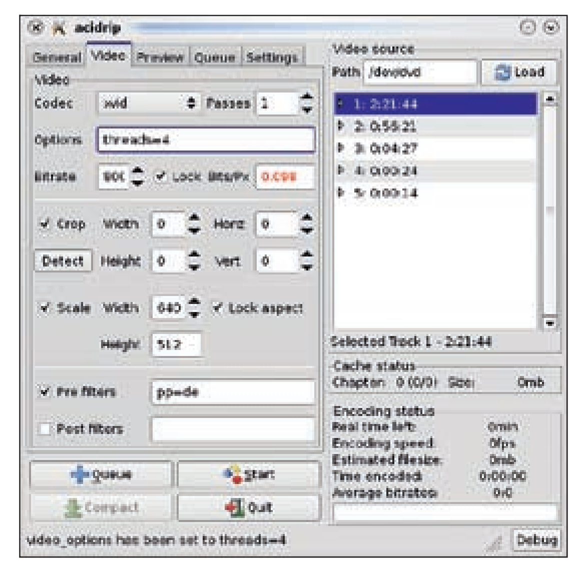
В Недавно я начал перемещать мою коллекцию DVD на ПК для хранения в NAS в формате Xvid. Я использовал Ubuntu 8.04 и программу *AcidRip*, которая, как я понял, основывается на *Mencoder*.

Однако добыча фильмов с диска в родном разрешении в два прохода с максимальным количеством битов на пиксель, рекомендованная *AcidRip*, отнимает четыре с половиной часа на кодирование полуторачасового DVD-трека (у меня одноядерный 3-ГГц P4). Это настолько медленно, что я задумался об обновлении до многоядерной машины, но мой любимый (после этого) племянник успел подарить мне Dell XPS710 с четырехъядерным процессором, на который я установил Ubuntu 9.04. Теперь на перекодирование того же фильма требуется 2 часа 15 минут.

Но когда я посмотрел на загрузку процессоров в системном мониторе, то увидел, что одно ядро вкалывало более чем на 99%, а другие три были заняты едва-едва на 0–5%. Занятое ядро периодически что-то скидывало на одно из остальных, через произвольные промежутки времени величиной в 10–50 секунд. Что происходит, и как мне сделать, чтобы все ядра работали одновременно?

Джефф Лайнс [Geoff Lynes]

О Использование нескольких ядер процессора подразумевает наличие в ядре Linux включенной поддержки SMP (симметричной мультиобработки, symmetric multiprocessing) и умение установленных программ задействовать более одного ядра одновременно. Первое условие у Вас выполнено, поскольку системный монитор показывает, что Linux распределяет процессы между ядрами процессора (это выравнивает температуру). Один процесс может быть запущен только на одном ядре – а несколько ядер полезны потому, что обычно в системе одновременно работает несколько программ, и каждая получает свою долю времени выделенного ей процессора.



› *AcidRip* может использовать многоядерные CPU, но это нужно явно указать в настройках каждого кодировщика.

Некоторые программы поддерживают многопоточность, когда работа расщепляется на ряд задач помельче (или потоков), каждая из которых использует отдельный процессор. *Mencoder* поддерживает многопоточность для большинства методов кодирования, но эту функцию требуется активировать. В большинстве кодировщиков это делается с помощью опции Threads. При настройке кодека, битрейта и тому подобного во вкладке Видео [Video] программы *AcidRip*, в поле Опции [Options] нужно выставить:

```
threads=4
```

для использования всех четырех ядер. Конечно, можно указать и меньшее число, чтобы оставить резерв ядер для других нужд компьютера. Чтобы не вводить опцию каждый раз, есть другая возможность: внесите данную настройку в конфигурационный файл *Mencoder*. Это **mencoder.conf** в каталоге **~/.mplayer**. Но нужно не просто вписать ее в файл – поскольку это подопция для каждого из кодировщиков, сделайте так:

```
lavcopts=threads=4
xvidencopts=threads=4
x264encopts=threads=4
```

Таким образом Вы установите значение 4 для опции threads кодировщиков Lavc, Xvid и x264, поэтому программа будет использовать до четырех потоков независимо от метода, выбранного в *AcidRip*, и необходимость всякий раз выставлять настройку отпадет. **МС**

4 Linux на флэшке-2

В Я попытался следовать инструкции, данной в номере LXF124 в ответе про Linux на флэшке. Я не совсем понимаю сообщения, появляющиеся при моих попытках выделить файлы для скачивания на www.vbox.me, но они, похоже, намекают, что ничего не работает — и успеха я не добился.

Алан Кокс [Alan Cox]

О Вы не предоставили нам почти никакого материала для размышлений: ни сообщений сайта, ни описания симптомов, сопутствующих Вашей неудаче. Тем не менее, похоже, что в пакет **Pocket-VirtualBox** включены компоненты *VirtualBox*, что нарушает лицензию Sun (двоичный пакет *VirtualBox* не совсем свободный, и его нельзя пересобрать и распространять по собственному желанию). Теперь он не содержит кода *VirtualBox*, и при запуске либо скачивает установщик *VirtualBox*, либо использует ранее загруженный дистрибутив приложения. Процесс инсталляции аналогичен, но при старте программы *Portable-VirtualBox* перед началом самой установки происходит скачивание необходимых файлов, так что необходимо рабочее сетевое соединение. Другой вариант – загрузить файлы с www.virtualbox.org и сохранить их на флэшку или аналогичный носитель; но надо будет убедиться, что версии *Portable-VirtualBox* и скачиваемого *VirtualBox* совпадают.

После завершения закачки и распаковки запустится *Portable-VirtualBox*. Выйдите из программы и сможете затем скопировать каталог **Portable-VirtualBox** целиком на флэшку и запустить программу уже оттуда.

Альтернатива – настроить Ваши виртуальные машины во время работы *Portable-VirtualBox*, запущенного с жесткого диска, и скопировать каталог уже после окончания настройки. **ГМ**

5 Пути файлов

В Я люблю запускать программы из консоли (например, из *Gnome Terminal* или *Xterm*). Многие классические приложения используют стартовый каталог как рабочий. Например, введя ls, я получаю данные о директории, в которой нахожусь; а если я введу emacs и попробую от- »

крыть файл, *Emacs* сочтет, что мне нужен файл, находящийся в том же каталоге.

Так-то так, но многие программы с графическим интерфейсом не следуют этой логике. Независимо от каталога, из которого я стартую *Inkscape* или *OpenOffice.org*, они используют другие директории – например, те, где файлы сохранялись ранее. Но мне этого не нужно, ведь я уже работаю в других сессиях, и мне каждый раз приходится продираться через долгую серию щелчков мыши, чтобы сохранить свою работу в текущем каталоге.

Это настолько нудно, что я решил написать вам. Как заставить приложения всегда использовать в качестве базового каталога тот, из которого я запускаю программу?

Г. Стоет [G Stoet]

О Вкратце можно сказать, что глобально повлиять на такое поведение нельзя. Если программе предписано открывать файловый диалог по последнему использованному пути, то этого не изменить без правки исходных текстов и перекомпиляции приложения. Некоторые программы используют пути по умолчанию, устанавливаемые в настройках. Другие поступают так, как Вы хотите – каталогом по умолчанию для *Inkscape* будет тот, из которого его запустили в консоли. Когда программа вызывается через иконку на рабочем столе или пункт меню, понятие текущего каталога отсутствует, и подбирается приемлемый вариант по умолчанию. Если Вам мешает именно это – как правило, настроить стартовый каталог для программы можно в редакторе меню графической среды Вашего рабочего стола. В KDE это называется Рабочим путем [Work Path] и находится на вкладке Дополнительно [Advanced] редактора меню [в KDE 3.5 данная настройка находится в подменю Личные файлы (Домой), и никакой вкладки Advanced там нет, – *прим. пер.*]. У редактора меню Gnome такая возможность отсутствует, поэтому понадобится небольшой сценарий оболочки, который выполнит **cd** в правильный каталог перед стартом программы; затем настройте меню на запуск скрипта вместо самого приложения.

Есть и обходные пути: например, передача имени пустого документа в текущем каталоге, хотя это уже считается «фокусом». Если Вам досадила какая-то конкретная программа, то «второй по качеству» подход – заполнить заявку на расширение функциональности в ее системе отслеживания ошибок (как правило, ссылка на нее присутствует на официальном сайте программы). А успешнее всего, видимо, будет самому залатать код и затем заполнить заявку, приложив свою заплатку; правда, такое не всем по зубам. Реальная возможность получить поддержку или даже помощь в воплощении идеи – поднять вопрос на форумах или в списке рассылки. НБ

6 Перенаправление портов

В Компания, на которую я сейчас работаю, заблокировала некоторые исходящие порты с большими номерами, и поэтому я не могу использовать *Putty*, чтобы зайти на мою домашнюю машину, с работающим на 9900 порту SSH-сервером. Можно ли приказать *Putty* туннелировать 9900 порт? Некоторые мои задачи зависят от SSH, запущенном на этом порту, поэтому поменять порт не является для меня подходящим решением.

Pk_fox, вопрос с форумов

О Если Вы хотите продолжать работать на эту компанию, то Вам не следует хитрить с политиками ее брандмауэра. Как минимум, нужно будет обратиться к администратору и узнать, можно ли Вам это сделать и какие порты доступны.

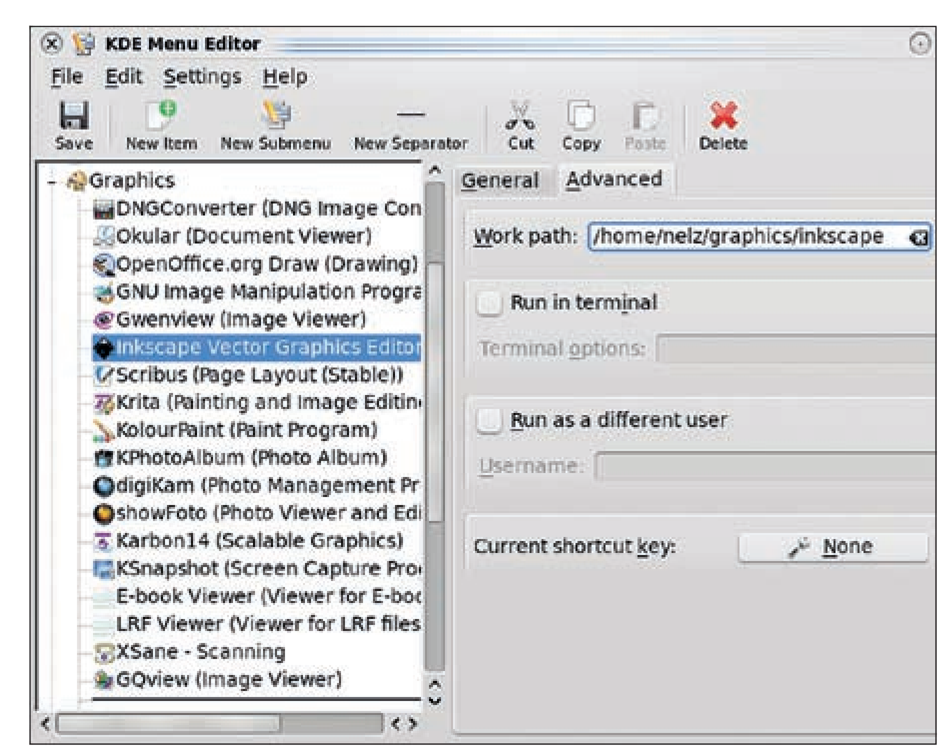
Проброс портов в *Putty* – не выход, поскольку на этом этапе Вы все равно останетесь за брандмауэром компании, и хотя в настройках можно указать подключение к другому порту, пользы от этого не будет. Полностью Вы контролируете только свой сервер, поэтому применяйте изменения там. Для этого есть пара способов, в зависимости от настроек Вашего сетевого подключения дома. Если Ваш маршрутизатор поддерживает проброс с изменением номера порта, то можно просто настроить его на перенаправление входящих соединений на приемлемом порту на порт 9000 на SSH-сервере. Также можно настроить маршрутизатор на перенаправление входящих соединений по порту 9000 туда же, если Вам нужен внешний доступ на этом порту. Если же у маршрутизатора такой функции нет, то можно настроить на прослушивание подходящего порта сам сервер.

Для этого нужно отредактировать **/etc/ssh/sshd_config** и изменить настройку Port. Поскольку нам все еще нужен доступ к порту 9000, то добавим новую строку – демон *sshd* может слушать несколько портов одновременно. Например:

```
Port 9000
Port 69
```

Порт 69 используется TFTP для загрузки по сети, и если у Вас на этой машине нет сервера сетевой загрузки, Вы можете использовать его, при условии, что соединение с ним разрешено Вашим брандмауэром. Если Вы перешли с порта 22 на 9000 в первую очередь для предотвращения попыток взлома по SSH, то очевидно, что использования стандартного порта нужно по возможности избегать. При вынужденном использовании порта 22 можно прибегнуть к чему-то вроде *sshutout* (www.techfinesse.com/sshutout/sshutout.html) – программы, блокирующей авторизацию при многократных неудачных попытках.

Еще один способ – туннелирование SSH, чтобы подключить один порт к другому, на той же машине или на удаленной, но для Вашей ситуации это уже будет перебор. Сервер, слушающий два порта, выглядит лучшим решением. ПХ



› Редактор меню KDE позволяет указать стартовый каталог программы.

7 Экран скукожился

В Я только что установил настольный вариант Ubuntu 9.04 на ноутбук Toshiba TE2000. Ubuntu работает хорошо, а экран – плохо. При разрешении 800×600 рабочий стол занимает лишь три четверти экрана. Все выглядит усохшим, включая панели меню и сами окна. Я добавил значения 1024×768 в xorg.conf, чтобы это разрешение появилось в списке доступных, перезагрузил ноутбук и обнаружил, что экранное изображение по-прежнему «усохшее», но увеличилось в масштабе, так что виден только угол; а когда я поместил курсор к краю экрана, он просто передвинул картинку.

Томас Барретт [Thomas Barrett]

О Все описанное Вами – классические приметы X-сервера, неверно определившего дисплей. Сервер думает, что Ваш экран поддерживает лишь разрешение 800×600, и устанавливает его как максимальное. Жидко-кристаллические мониторы используют одну ячейку экрана на пиксель (иначе масштабированный экран выглядят жутко клетчатым), и в итоге Вы пользуетесь серединкой 800×600 от своих 1024×768. X-сервер поддерживает размеры экрана большие, чем размеры физического дисплея, как если бы Ваш монитор был просто окном к рабочему столу. Именно это Вы и видите, когда выбираете больший размер экрана: 800×600 пикселей рабочего стола, который охватывает 1024×768 пикселей, перемещается помощью мыши.

X-сервер получает информацию о дисплее от графической карты и монитора посредством EDID (Расширенные данные идентификации дисплея, Extended Display Identification Data), и посылаемую информацию можно увидеть, установив пакет **read-edid** и набрав в консоли:

```
sudo get-edid | sudo parse-edid
```

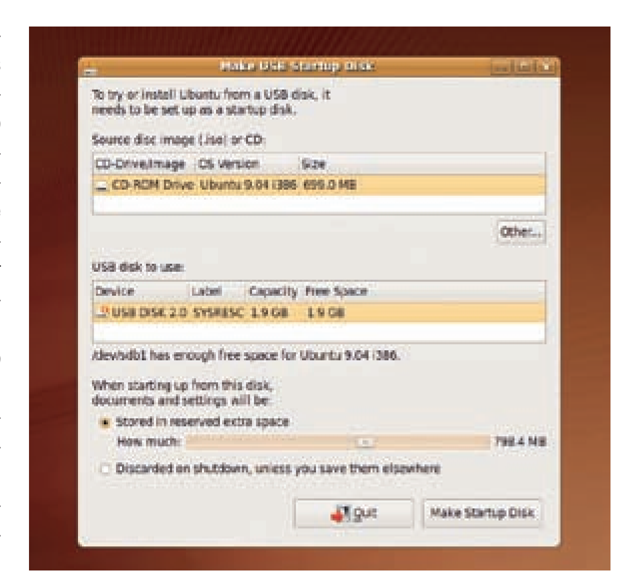
Команда **get-edid** считывает информацию и передает ее **parse-edid** для перевода в удобочитаемую форму.

Чтобы отменить эти настройки и как-то сообщить X-серверу, что экран Вашего ноутбука способен поддерживать дисплей размером 1024×768 пикселей, откройте **/etc/X11/xorg.conf** с правами root и найдите раздел Monitor. По умолчанию он содержит только идентификатор; следовательно, все остальное настраивается автоматически. Прописав все вручную, мы отменяем автоматическую настройку, так что убедитесь, что параметры действительно поддерживаются Вашим монитором: в данном случае 1024×768 – правильное разрешение. В этот раздел надо будет добавить строку Modeline, но чтобы рассчитать значения, придется попотеть (поэтому по умолчанию и используется EDID).

Существует онлайн-калькулятор таких значений, доступный по адресу http://bit.ly/C2d55, и выбрав стандартное разрешение в 1024×768@60Гц и нажав кнопку Calculate, мы получим следующее:

```
Modeline "1024x768" 65 1024
1048 1184
1344 768 771 777 806 -hsync
-vsync
```

Лучше всего запустить этот тест на сайте самому, чтобы все подробности совпадали со спецификациями Вашей аппаратуры; затем скопируйте выданный код в раздел Monitor файла **xorg.conf** (сразу после строки идентификатора). Перезапустите X-сервер или перезагрузитесь, и тогда Вы должны получить разрешение 1024×768. Перед редактированием **xorg.conf** сделайте его резервную копию, чтобы иметь возможность отката к старым настройкам, если новая конфигурация не сработает. ПХ



› Программа-генератор стартовых дисков создаст USB-версию Ubuntu Live-CD для мобильности или для применения с нетбуками.

8 Флэшовая Ubuntu

В Я буквально годами пытался заставить ОС загружаться с флэшки. Не будучи асом в технологиях, я искал помощи в разных источниках, но безуспешно. Увидев статью «Кар- »

Часто задаваемые вопросы

Печать

» Что такое CUPS?

Common Unix Printing System (CUPS) – это набор драйверов и утилит, предоставляющих полное обеспечение работы печатающего устройства, управление и использование его в Linux и других Unix-подобных системах.

» Значит, это драйвер для принтера?

Гораздо больше. CUPS предоставляет «portable printing layer» – переносимый слой печати между приложениями и принтером; он включает не только драйверы, но также и все, что нужно программам, чтобы иметь возможность выводить что-то на печать.

» Надо ли мне вводить команды в консоли и редактировать конфигурационные файлы, чтобы мой принтер начал печатать?

Вовсе нет. У CUPS есть графические утилиты настройки, работающие через браузер. Наберите в нем адрес http://localhost:631. Возможно, вам понадобится ввести пароль вашего пользователя или пароль root, после чего вы увидите домашнюю страничку CUPS. Здесь можно добавлять/удалять принтеры, управлять текущими заданиями печати и читать документацию.

» Зачем нужен браузер вместо стандартной графической программы?

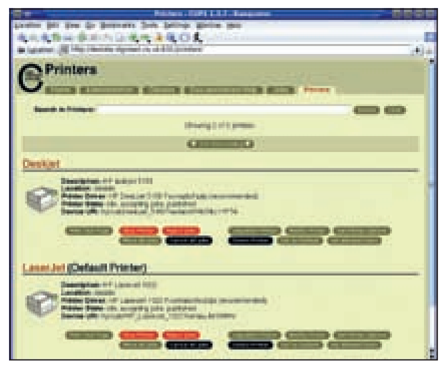
Web-интерфейс означает, что не нужно устанавливать никакого графического инструментария; можно даже не запускать X-сервер. Доступ будет и через текстовый браузер, и даже с другого компьютера.

» Но ведь это не безопасно?

Настройки по умолчанию разрешают подключение только с локального хоста. Это можно изменить, разрешив соединения из локальной сети (разрешать интернет-подключения, как правило, не рекомендуется). Можно контролировать действия пользователей – например, позволить им управлять своими заданиями на печать, но не удалять принтеры.

» Как во все это вписывается *Gimp-print*, и что общего у *GIMP* с печатью?

Это набор драйверов для принтеров, разработанных для использования в *GIMP*. Хотя *GIMP* хорошо ладит с CUPS, некоторые принтеры выдают лучший результат при использовании драйверов *Gimp-print*. Эти драйверы теперь работают и с CUPS; можно считать, что *Gimp-print* – расширенный набор драйверов, работающих со всеми приложениями, которые выполняют печать через CUPS. Проект сейчас переименовали в *Gutenprint*, чтобы избежать путаницы с *GIMP*.



› Управляем принтерами через web-интерфейс с CUPS.

» Как мне узнать, поддерживается ли модель моего принтера?

Списки поддерживаемых принтеров и рекомендованных для них драйверов ищите на www.linuxprinting.org.

манный Linux» в LXF113/114, я взликовал. Я очень старался и следовал инструкциям, но все равно ничего не вышло. Помаленьку я решал все проблемы, пока не дошел до последнего шага, где утверждается, что после выбора языка загрузочная флэшка готова и я попаду в Live-сессию. Но у меня так никогда не случалось. После выбора английского языка я получаю запрос с Live CD: хочу ли я попробовать, не трогая жесткого диска, установить, протестировать память и так далее. Я перепробовал разные Live-диски и даже ISO-образ — и все безрезультатно.

Лоуренс Джонстон [Lawrence Johnston]

О Если Вы следовали инструкциям для Ubuntu, то и должны были это увидеть, поскольку там создается USB-версия Live CD. Сообщения, которые Вы получили – это загрузочное меню, точно такое, какое появляется при старте с CD. Выбирайте первый пункт – тогда загрузится рабочий стол, и Вы попадете в Live-среду как Live-пользователь.

При желании можно пойти еще дальше: спрятать меню и автоматически загружаться в Live-среду. Смонтировав флэшку, отредактируйте **syslinux/syslinux.cfg**. Найдите строку с параметрами таймаута. Скорее всего, там выставлено число 300, что означает задержку в 30 секунд. Удалите ее и замените следующими строками:

```
menu hidden
timeout 30
```

Меню спрячется, а таймаут сократится до 3 секунд – этого достаточно, чтобы нажать Esc, если меню все-таки понадобится. Сохраните файл и отмонтируйте флэшку, но не вынимайте ее из разъема. Теперь запустите в консоли команду:

```
sudo syslinux /dev/sdXY
```

где **sdXY** – устройство, соответствующее Вашей флэшке. Вероятнее всего, это будет **/dev/sdb1**, при наличии единственного жесткого диска на **/dev/sda**, но проверить это можно, запустив в консоли с подмонтированной флэшкой команду **df**. Теперь при загрузке с флэшки после короткой паузы Вы попадете прямиком в загрузочный экран Ubuntu. **НБ**

9 Медиа-захват

В У меня есть устройство видеозахвата Compro Videomate C200 USB AV, которое я хочу применить в Ubuntu Linux. Возможно, я избалован хорошей поддержкой Plug and Play в Linux, но, к сожалению, с этим устройством она не сработала. При подключении я получаю следующие сообщения от *dmesg* и *lsusb*:

```
dmesg
[ 1757.672055] usb 1-3: new high speed USB
device using ehci_hcd and address 3
[ 1757.845960] usb 1-3: configuration #1
chosen from 1 choice
lsusb
```



› **Не все, что лезет в USB-разъем, может быть опознано Linux. По крайней мере, сразу.**

```
Bus 001 Device 003: ID 185b:0200 Compro
```

По-моему, это означает, что устройство распознается, но в программах типа *Audacity* оно не появляется в списках устройств захвата. Хотя на данном этапе я пытался работать только со звуком, но надеюсь, что ваши советы помогут разрешить проблему и с видеозахватом.

Нет ли утилиты, которую я мог бы использовать, чтобы мое устройство заработало, или хотя бы консольного способа?

Стюарт Тилли [Stuart Tilley]

О Вывод *lsusb* говорит о том, что устройство не распознано. Два числа ID означают коды изготовителя и модели, а текст, следующий далее, говорит, что системе известен только код изготовителя, поэтому и нет никакой информации о модели (код изготовителя известен потому, что в базе данных USB ID есть устройство DVB-t того же самого изготовителя). Если система не опознает это устройство, значит, не может и подключить драйверы, даже если они есть и установлены.

Здесь Ваши возможности ограничены, поскольку поиск в Интернете не дал никакой полезной информации. Первой вехой на пути у Вас должен быть сайт www.linux-usb.org, где можно проверить свежие поступления в базу данных идентификаторов «Изготовители/устройства», использовав поиск по изготовителю и коду продукта. Если и после этого устройство не будет опознаваться, пошлите подробную информацию в список рассылки linux-usb-users. Очень может быть, что Ваше устройство использует чипсет, уже поддерживаемый модулем ядра, и дело стало только за тем, чтобы добавить соответствующую информацию в базу данных USB. Также можно попробовать связаться с изготовителем и попросить помощи у него.

Тем не менее, вполне может оказаться, что поддержка данной конкретной модели не планируется, и простейшим выходом тут будет ее замена на другое устройство, поддерживаемое в Linux. В этом случае неплохо бы написать изготовителю, что Вы приобрели другое устройство у их конкурента, потому что оно совместимо с Linux.



Коротко про...

Выполнение команд пакетом

При использовании консоли бывают случаи, когда нужно запустить последовательность команд (такое может случиться и при работе с графическим интерфейсом, но тогда решение будет отнюдь не таким простым). Классическим примером тут служит стандартный способ установки программ из исходных текстов, требующий запуска **./configure** (возможно, с аргументами), а затем **make** и **make install**. Выполнение некоторых из этих шагов может длиться часами, в зависимости от сложности кода и возможностей вашей машины. Ожидать окончания одного шага, чтобы ввести следующий, неэффективно – лучше сделать так:

```
./configure; make; make install
```

Точка с запятой говорит о том, что команды должны выполняться последовательно, как если бы вы вводили их по очереди. Но тут можно заметить потенциальную проблему: а ну как **./configure** или **make** завершатся с ошибкой? Будут ли выполнены последовательно заданные команды? Ответ – да, причем они наверняка скроют сообщения об ошибках, и вы не узнаете о проблемах, пока не возьметесь работать с этой программой.

Вот более практичный способ запуска данных команд:

```
./configure && make && make install
```

&& – это логический оператор. Приведенная последовательность означает «если **./configure** истинно, и **make** истинно, и **make install** истинно». Оболочка определяет истинность команды, запустив ее на предмет выдачи возможных ошибок. При их наличии в первой команде следующая запускаться не станет, поскольку тест уже провален, так что объединение команд с помощью **&&** запускает их поочередно, но останавливается, как только одна из команд возвращает ошибку – а значит, вам незачем неотлучно дежурить у консоли. Сопутствующий оператор – ||, означающий «или», то есть при вводе:

```
команда1 || команда2
```

команда2 сработает только в том случае, если **команда1** завершится с ошибкой.

Это не очень ценно в интерактивной оболочке, но часто применяется в скриптах, например:

```
некаякоманда1 || echo «Что-то сломалось!”
```

Большой вопрос Можно ли работать с электронными книгами в Linux?

10 E-книги и Linux

В Мне подарили Sony E-Reader, который продается только с программами для работы в Windows. Я пытался смонтировать его как USB-устройство и копировал на него электронные книги, но в меню их не оказывалось. В комплекте шел CD с сотней электронных книг, но я не знаю, как их прочитать. Мне придется установить Windows, или все-таки с ним можно работать в Linux?

Пит Хардинг [Pete Harding]

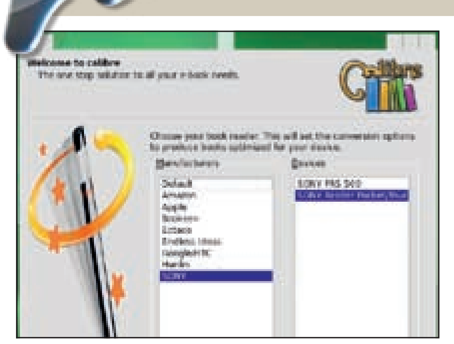
О Вы не указали конкретную модель вашего устройства, но мы использовали PRS-505, и варианты Pocket и Touch должны работать так же. Что касается передачи материала, защищенного DRM (это относится к книгам, купленным онлайн), то для разблокирования DRM-защиты во время установки книг на устройство чтения понадобится ПО Windows. Эти программы не работают из-под *Wine*, и потребуется полноценная установка Windows – либо в виде двойной загрузки, либо в виртуальной машине *VMware* или *VirtualBox*. Если вам нужно перенести на электронную книгу материалы, не защищенные DRM, например, книги из «проекта Гутенберга» (Project Gutenberg) или же свежие номера Linux Format в формате PDF, приобретенные через наш сайт, это можно сделать и в Linux.

При подключении электронной книги она определяется как USB-накопитель и может быть смонтирована. Тем не менее, разделов на этом устройстве нет, поэтому мы монтируем его целиком, как дискету, в виде **/dev/sdb**, а не **/dev/sdb1**. Хотя устройство и можно смонтировать, просто скопировать файлы и затем читать их нельзя, тут нужно управлять базой данных электронной книги. Вам поможет программа *Calibre* (http://calibre.kovidgoyal.net). Добавление книг на устройство состоит из двух этапов: сначала добавим книги в библиотеку *Calibre*, затем выберем те, которые хотим переместить в читающее устройство. Библиотека книг на вашем компьютере может вместить объем больший, чем устройство, а на него вы отправите только книги, интересующие вас в данный момент. *Calibre* также позволяет читать книги с компьютера.

Перед запуском программы подключите и смонтируйте устройство (это вроде бы нужно для его правильного опознания). Первый запуск программы вызовет мастер настроек, затем вы увидите пустую библиотеку. Щелкните по иконке Добавить книги [Add Books] и выберите несколько или все книги с прилагающегося CD, тем самым добавив их в библиотеку *Calibre*. Книги, содержащиеся на диске, уже имеют правильный формат, но материалы, полученные из других источников, могут нуждаться в конвертации. Это можно сделать вручную, с помощью кнопки Преобразовать книги [Convert E-Books], а если пропустить этот шаг, перед процессом копирования книги преобразуются автоматически. Выберите нужные книги и нажмите кнопку Переместить на устройство [Send to Device]. Через пару секунд перемещение завершится, и можно будет нажать кнопку Извлечь [Eject] рядом с иконкой вашего устройства перед тем, как отмонтировать и отключить вашу электронную книгу. Свежескопированные книги теперь должны быть видны на устройстве.

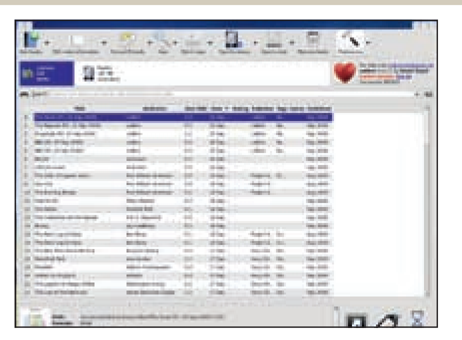
Calibre также настраивается на автоматическую загрузку и конвертацию новостей из сети. Если е-книга подключена, *Calibre* даже перенесет в нее новости – наутро устройство будет и полностью заряжено, и набито новостями.

Шаг за шагом: Читаем электронные книги с Calibre



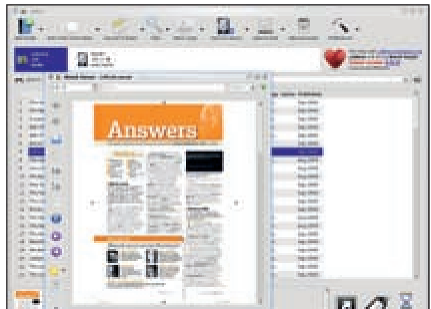
1 Начальная настройка

Во время первого запуска *Calibre* должна определить вашу электронную книгу (при условии, что она смонтирована) и все для вас настроить.



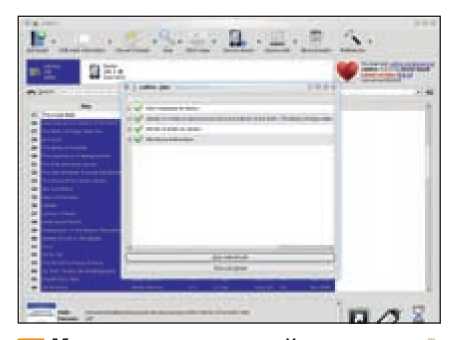
2 Полная библиотека

Calibre поддерживает библиотеку е-книг на жестком диске, и первым вашим шагом должно быть добавление книг с CD.



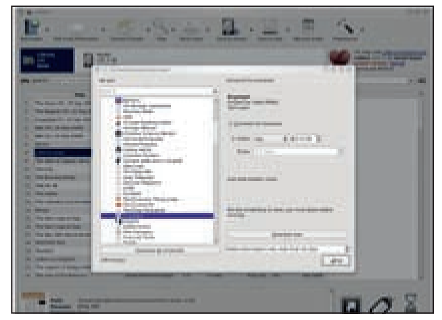
3 Читаем в Calibre

Книги можно читать и в *Calibre*, что очень удобно для проверки конвертации, но не для этого же вы приобретали свое устройство!



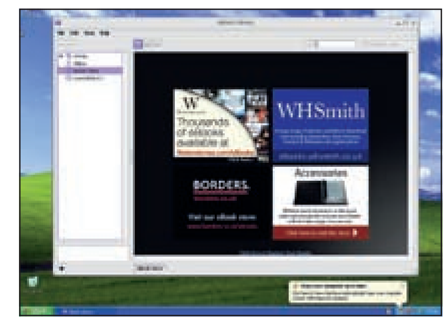
4 Копируем в устройство

Calibre скопирует выбранные вами книги в устройство, попутно позаботившись о необходимой переконвертации.



5 Еще и новости

Настроим расписание для загрузки и конвертации новостей из различных RSS-лент и превратим нашу электронную книгу в утреннюю газету.



6 Не без MS...

Найти свободную версию программы, разблокирующей защиту DRM, вряд ли вообще возможно, поэтому для чтения коммерческих книг не обойтись без Windows. НБ

Лучшие в мире новинки свободного ПО

LXFHotPicks

Энди Хадсон
Когда Энди не притворяется, что отлаживает почтовые сети, он ныряет среди коралловых рифов в поисках жемчужин для HotPicks.

Читалка RSS

RSSOwl

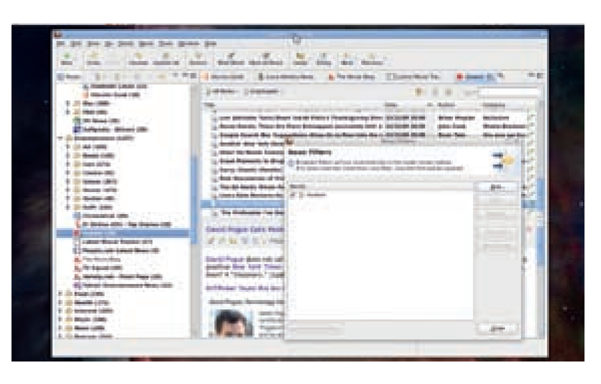


Версия 2.0 **Сайт** www.rssowl.org

Мы в Башнях **LXF** обольщаемся, что находимся в курсе всех текущих событий; на нашу учетную запись Google Reader приходят новости со 140 различных новостных лент со всего мира, и отслеживание событий уже превратилось в повинность. Вдобавок и интерфейс не самый лучший, поскольку с ним все наши ленты начинают зависеть от качества интернет-соединения. Порой нам недостает достойного мультиплатформенного RSS-клиента, пригодного для использования на рабочем столе и установки его везде, где бы мы ни оказались. Именно таким и пытается стать *RSSOwl* – Java-приложение с массой функций, облаченных в аккуратный и внятный интерфейс.

Установка и запуск просты: щелкните дважды по двоичному файлу – и вперед. Как и любая другая программа, собирающая новости, она полезна ровно настолько же, насколько ленты новостей, ради которых вы ее настраиваете, а выбор параметров по умолчанию в *RSSOwl* огромен. Если вы пользуетесь Google Reader или Bloglines, то, скорее всего, уже организовали свои ленты в удобную подборку. Экспортируйте ее в файл OPML, и *RSSOwl* «прожует» его и отобразит новости в привычном для вас виде.

> «Новостные ленты можно искать прямо из интерфейса.»



› Подобно Сове из рассказов о Винни-Пухе [Owl — Сова, — *прим. пер.*], *RSSOwl* — просто кладезь информации и знаний.

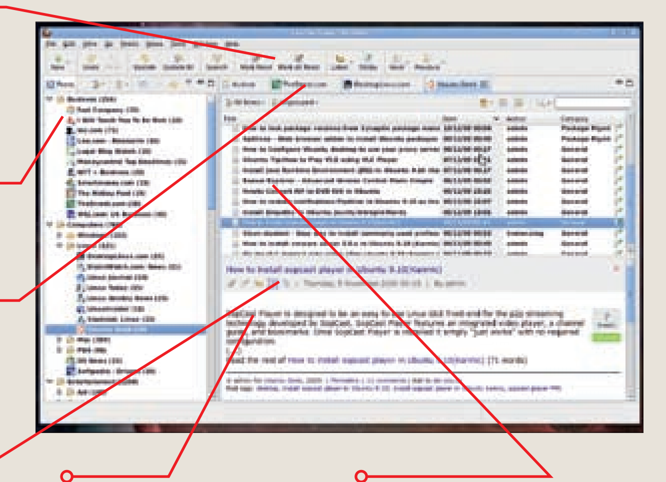
Поиск лент

Больше всего нам понравилось, что новые ленты можно искать прямо из интерфейса программы. Поиск Google Android предоставил нам на выбор свыше 60 результатов. Часть была притянута по ошибке – в них Android просто упоминался в статье; но большинство попало в яблочко. Поиск TuxRadar вернул примерно столько же источников, и все они очень легко объединились в новую группу в *RSSOwl*.

RSSOwl сообразительно замечает, что вы собрались импортировать ленту, на которую уже подписаны, и скрывает ее, чтобы она не добавилась повторно. Вам не придется и возиться с бесконечной чередой списков: *RSSOwl* сгруппирует подборки, распределив ваше чтиво по рабочим, личным и прочим интересам. Можно также создавать поисковые фильтры для входящих статей и помещать определенную информацию в настраиваемые «новостные корзины» (News Bins); главное преимущество для вас при этом – глубокая персонализация, поскольку вы получите статьи, отвечающие вашим интересам.

Мы не могли остаться равнодушными к *RSSOwl* – он дает отличный интерфейс для управления новостями RSS и облегчает поиск новых новостных лент. Если управление вашими лентами стало нудной рутиной, срочно обзаводитесь *RSSOwl*, чтобы увидеть, как это делается в лучших домах.

Исследуем интерфейс RSSOwl



Навигация
Используйте кнопки на панели инструментов: они помогут вам осуществлять навигацию по множеству новостных RSS-лент.

Список лент
Здесь перед вами иерархия всех категорий и новостей.

Просмотр во вкладках
Когда вы выбираете очередную новостную ленту, открывается вкладка, в которой отображается ее содержимое.

Несите весть
Вы можете поделиться статьей, используя кнопку Share News, чтобы отправить ее в Twitter, Facebook и т. п.

Закладки
Нажмите сюда, чтобы добавить в ваш альбом социальную закладку и поделиться ею со всем миром.

Статьи
Здесь вы найдете самые последние статьи из выбранных вами источников, с подробной информацией по каждому посту.

Программа создания резервных копий

Déjà Dup

Версия 11.0 **Сайт** https://launchpad.net/deja-dup

Приложения нечасто заставляют нас от души порадоваться. Обычно-то все наоборот: пока скомпилируешь новейшую программу, желая осчастливить читателей, слезами умоешься. Тем не менее, хорошее тоже случается, и конкретно это произошло в случае *Déjà Dup*, приложения для создания резервных копий, в интерфейсе которого всего две кнопки: Скопировать [Backup] и Восстановить [Restore].

Функция у *Déjà Dup* очень простая: скопировать ваши данные в другое место либо восстановить их снова на вашем компьютере. Но прежде чем перейти к этой стадии, вам придется скомпилировать программу, для чего обзаведитесь *Duplicity*, а также *Unique* и *Gnome-keyring*. Загрузив эти пакеты, можно приступать к делу.

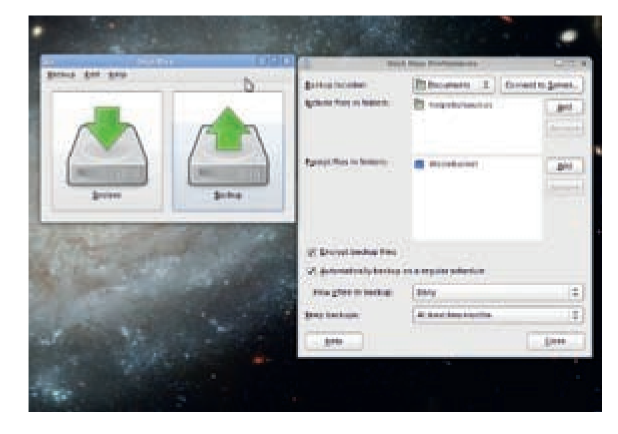
Хотя мы не нашли *Déjà Dup* в нашем меню Gnome, оно легко запускается через команду **deja-dup** в терминале. Тут перед вами и появятся те две заветные кнопки. Но сами по себе они ничего сделать не смогут: нужно еще указать им, что именно и где вы хотите сохранить как резервную копию.

Настройка

Настройка выполняется либо нажатием на кнопку Backup – она проведет вас через несколько экранов, которые помогут вам сконфигурировать создание резервных копий; либо редактированием настроек, чтобы задать параметры по умолчанию.

Déjà Dup поддерживает соединения через SSH, FTP и любой иной метод, понимаемый Gnome, что облегчает задачу резервного копирования на внешний носитель. Оно также совместимо с Amazon S3, и можно хранить резервные копии «в облаке». *Déjà Dup* позволяет шифровать данные, что мы и рекомендуем делать, если вы пользуетесь услугами глобальной сети.

> «Déjà Dup снимает проблемы копирования на внешний носитель.»



› Интерфейс минималистский, но *Déjà Dup* сочетает простоту с эффективностью.

Допукается также определять включенные и исключенные файлы и папки, указывать частоту резервного копирования (хотя, к сожалению, нельзя создавать резервные копии по запросу) и оптимизировать ведение журнала – по умолчанию он хранится вечно, но этот срок можно и сократить, если вопрос объема памяти является критичным.

Нам понравилось *Déjà Dup*, потому что оно снимает проблемы с резервным копированием на внешний носитель и значительно облегчает процедуру восстановления. Упрощение процесса резервного копирования должно являться целью любого приложения для создания резервных копий, и *Déjà Dup* определенно этой цели достигает.

Совместная работа над текстом

Gobby

Версия 0.4.11 **Сайт** http://gobby.0x539.de

Пардон за каламбур, но Google погнал волну по всей Сети, запустив Google Wave [wave – волна, – *прим. пер.*] – новую концепцию сотрудничества, позволяющую интерактивно работать через Wave с другими людьми всевозможными способами (**LXF123**).

Gobby, подобно Wave, обеспечивает сотрудничество в реальном времени, но только при работе над текстовым документом. Однако для создания соединения с другими пользователями *Gobby* вам понадобятся определенные порты в вашей сети, а не непосредственно в Интернете (как это делается в Wave).

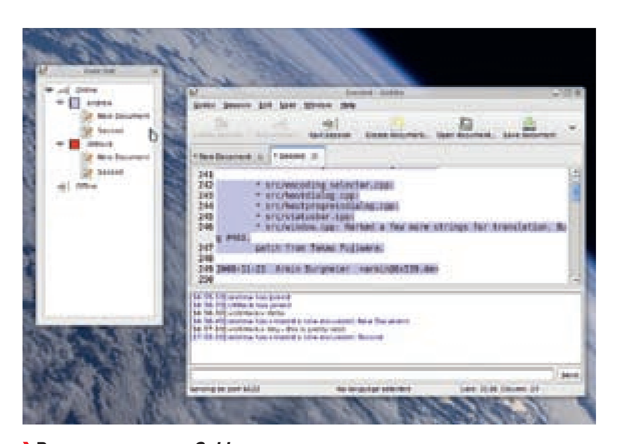
Создавая новую сессию, укажите порт для подключения других пользователей; следите за его правильностью, если вы осуществляете переадресацию (проброс) портов.

Можете задать пароль, чтобы исключить вторжение посторонних. Затем открывается соединение и появляются две панели: верхняя – сам документ, а нижняя служит интерфейсом для журнала и чата, чтобы вы могли постоянно отслеживать все, что происходит.

На рабочей волне

Каждому пользователю присваиваются уникальные имя и цвет, и по ходу внесения изменений выделенный определенным цветом текст вставляется в документ, чтобы был виден автор правки. В локальной сети изменения происходят мгновенно, и поначалу немного странно видеть, как несколько человек набирают текст в реальном времени.

Однако через пару минут вы привыкнете к тому, что над одним текстом трудятся несколько человек. Переписать то, что сделано другими, очень легко, и, похоже, в *Gobby* нет механизма для защиты отдельных частей содержания текста. Зато имеется выделение синтаксиса для целого ряда языков.



› В панели журнала *Gobby* отслеживает все изменения, внесенные пользователями в документ.

Участники в любой момент могут сохранить документ локально или создать новый; однако открывать новый файл придется самому – автоматически *Gobby* этого не делает. Открытые файлы отображаются во вкладках в верхней части экрана, но защитить паролем документы нельзя – только сессию сервера. Окно User List отображает, кто участвует в сессии, и напоминает, какой цвет присвоен каждому пользователю.

Клиент электронной почты

Claws

Версия 3.7.3 **Сайт** www.claws-mail.org



› *Claws* — старый фаворит, который часто отодвигают в сторону ради *Evolution*, *Kmail* или *Thunderbird*.

Любой неслучайный пользователь Linux рано или поздно сталкивается с *Claws*, бывшим некогда *Sylpheed Claws*. Это – клиент электронной почты, существующий с незапамятных времен, и у него есть верные поклонники. В последний раз мы рассматривали его в LXF94, так что, принимая во внимание недавний релиз, самое время вновь вернуться к этой классической программе.

Изначально своим успехом *Claws* обязан тому, что его разработчики умудрились создать по-настоящему шустрый клиент электронной почты с большим набором функций, и произошло это до того, как на арену, поигрывая мускулами, вышел *Thunderbird*. Основанный на библиотеках *GTK*, которые вам и потребуются, если вы не работаете в Gnome, *Claws* отличается замечательной простотой компиляции, которая требует всего лишь нескольких пакетов. Прочитав файл **INSTALL**, вы увидите, что вас ждет, поскольку при сборке программы предстоит выбирать из десятков опций. Кроме того, для расширения состава функций вашего *Claws* есть и список дополнительных пакетов. Неизменный триумвират **configure**, **make** и **make install** – это все, что стоит между вами и *Claws*. После завершения установки вы обнаружите новый значок, расположенный в меню Интернет рабочего стола, и можете приступать.

Как и в большинстве программ для работы с электронной почтой, при первом запуске вам будет предложено настроить свою учетную запись. Вам будут заданы обычные вопросы, но, к сожалению, здесь нет встроенного распознавания популярных сетевых почтовых служб вроде Google Mail, которое имеется в других программах для работы с электронной почтой. Понятно, что если вы компилируете собственные программы, то, скорее всего, уже настолько продвинуты, что наизусть знаете нужные настройки, но все же было бы неплохо иметь в *Claws* функцию автозаполнения некоторых полей на основании адреса электронной почты, который вы ввели в самом начале.

Интерфейс-аскет

Заполнив начальную учетную запись, вы попадете в интерфейс *Claws*. Тем, кто привык к *Evolution* или *Thunderbird* с *Lightning*, *Claws*, вероятно, покажется слегка минималистским. Он работает исключительно с электронной почтой и не включает календаря, но простенькая адресная книга все же есть.

Чтобы испытать *Claws*, мы напустили его на ящик, используемый у нас для отслеживания списков рассылки Fedora – на момент написания данного материала в нем было более 192 000 сообщений. *Claws* не тормозил ни секунды и лихо загрузил все письма порциями по 50 штук. Он также автоматически подобрал все IMAP-папки, которые мы ранее настроили в Google Mail другой программой. Это облегчило задачу применения встроенных фильтров, и не успели мы и опомниться, как все наши сообщения были отсортированы в соответствии с принадлежностью к спискам рассылки Fedora.

> «Есть список пакетов для расширения состава функций.»

Добавка функций

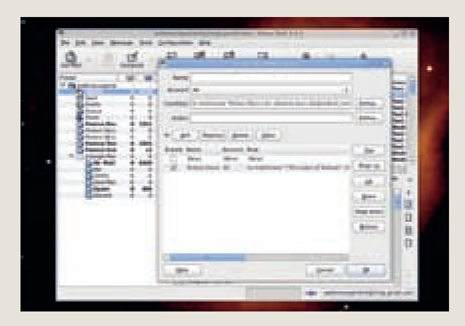
Одна из сильнейших сторон *Claws* – возможность расширения посредством модулей и опций; их полно на сайте *Claws*, и некоторые открывают доступ к основной функциональности *Claws* через такие языки, как Perl или Python. Версия 3.7.3 внесла массу усовершенствований интерфейса; два из них обучат вас корректно работать с электронной почтой. Во-первых, выдается предупреждение, если вы собрались отправить вложение, превышающее допустимый размер. Во-вторых, можно установить фильтр на наличие вложения, которое затем можно отбросить с помощью модуля расширения.

Claws – одна из тех жемчужин, о которых легко забыть, но его стоит попробовать, если вы ищете перемен.

Шаг за шагом: Настройка правил

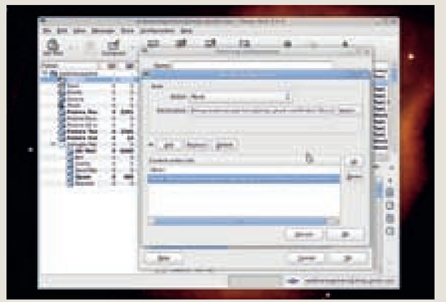
» Приступаем

Чтобы задать правила сортировки почты, перейдите в меню Tools [Сервис] и выберите Create Filter Rule [Создать фильтр].



» Опции

Claws позволит выбрать существующее сообщение и использует его данные для заполнения ряда полей – например, Subject (Тема) или From (Отправитель).



» Мотор!

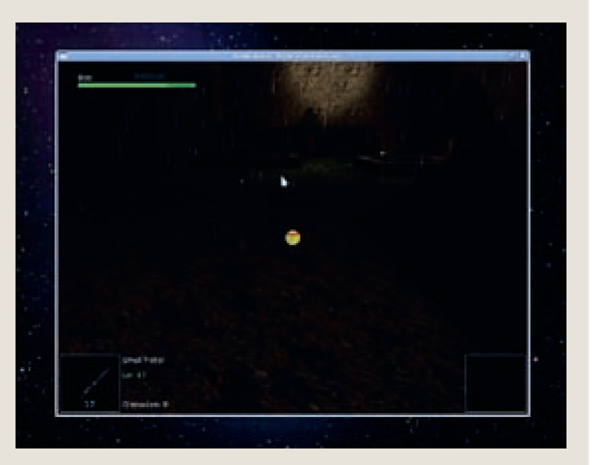
Правила фильтрации определяет тщательно продуманный синтаксис. Можно также создать правила для обработки данных, например, удаления вложений.

HotGames Развлекательные приложения

Стрелялка

Blob and Conquer

Версия 1.10 **Сайт** http://tinyurl.com/ktw2q6



› Палите по противникам в *Blob Wars II* — и увидите, как они рухнут к вашим ногам.

Мы затрудняемся описать *Blob and Conquer*: *Blob Wars Episode II*; игра выламывается из всех категорий когда-либо виденных нами стрелялок.

Сюжет таков: вы – небольшой пузырь по имени Боб [Bob], по форме напоминающий теннисный мяч с банданой в стиле Рэмбо, который карабкается с уровня на уровень с целью уничтожения зла в лице Galdov. Игра ломится от злодеев и препятствий, и, возможно, она самая жестокая из всех, с которыми мы сталкивались в последнее время.

У вас на вооружении дробовики, пистолеты и гранаты – а при случае и автоматы – и всю эту мощь вы огненным ливнем обрушиваете на головы противников. Управление осуществляется мышью и с клавиатуры, но можно также использовать совместимый джойстик.

Здесь есть несколько чудесных звуковых эффектов, сопровождающих отправку ваших противников на небеса, к огромному шару; а их бренные останки остаются на земле, там, где они упали.

Крутой пузырь

Графика не самая передовая, но один взгляд в лицо Боба скажет обо всем: он готов сразиться с самым крутым пузырем, и никто – в смысле, ни один пузырь – не загородит ему пути.

Для запуска *Blob Wars* вам понадобятся некоторые библиотеки SDL, включая *Mixer*, *Image*, *TTF* и *ZLib*. У тех, кто уже играл в игры на основе SDL, они, скорее всего, имеются.

> «Blob and Conquer выламывается из всех категорий стрелялок.»

Если компиляция из исходных текстов вас не радует, для некоторых дистрибутивов предусмотрены пакеты; процесс сборки отличается быстротой, и через некоторое время вы уже начнете поливать своих противников из Узи.

Единственное, что нас малость расстроило – исходная чувствительность мыши; но ее можно настроить в процессе игры.

Бомбардировщик

Bomns

Версия 0.99.2 **Сайт** http://greenridge.sourceforge.net



› *Bomns*, может, и не выглядит сложным, но его элемент стратегии изрядно напрягает ваши мозги.

Сотрудников **LXF** порой обвиняют в склонности к жестоким играм; нас извиняет то, что, создавая этот превосходный журнал, мы находимся в постоянном напряжении и нуждаемся в отдушине – чтобы пар выходил.

Недавно мы наткнулись на *Bomns*. Основанный на старой текстовой игре, сюжет весьма напоминает *Bomberman*, где два игрока бредут по полю, собирая разные штуки и пытаясь заодно вздуть противника.

Игра доступна в виде tar-архива и проста в установке; убедитесь, что у вас есть библиотеки SDL и *GTK*, потому что, хотя основа и текстовая, экран настройки требует наличия *GTK*. Установить и запустить игру легко, и через пару секунд компиляции вы уже готовы вместе со своим противником нырнуть в пучину боли и страданий – по крайней мере, идея такова.

Вы с вашим противником бродите по случайным образом сгенерированной сетке; в отличие от *Bomberman*, она здесь состоит не только из одиночных линий. У вас полная свобода перемещения.

Больше бомб

Поле битвы усеяно всякими приспособлениями, а также бонусами, которые следует подбирать: это, например, лишние бомбы или довольно полезный телепорт. Управление – сама простота: первый игрок использует клавиши со стрелками, второй – WASD; а нажатие Return или пробела метнет бомбу для каждого игрока.

> «Заставляет думать, как именно загнать врага в ловушку.»

Игровое поле настолько обширно, что заставляет усиленно думать, как именно загнать своего врага в ловушку – просто изолировать его огнем, как в *Bomberman*, нельзя, и именно этот элемент стратегии придает *Bomns* неповторимый блеск.

IRC-клиент

Konversation

Версия 1.2 **Сайт** http://konversation.kde.org

Возможно, наилучший способ связаться с легионами пользователей Linux во всем мире – это Internet Relay Chat, или IRC. Он также и оплот множества хакеров старшего поколения, которые использовали каналы IRC с незапамятных времен. Мы с нежностью вспоминаем свое общение в DALnet в середине 90-х, когда в IRC можно было попасть в основном через наше университетское соединение.

Наши радары недавно зафиксировали обновление клиента *Konversation* для KDE. Должно быть, в тот день боги Linux нам улыбнулись, потому что *Konversation* оживил наш интерес к IRC-каналам.

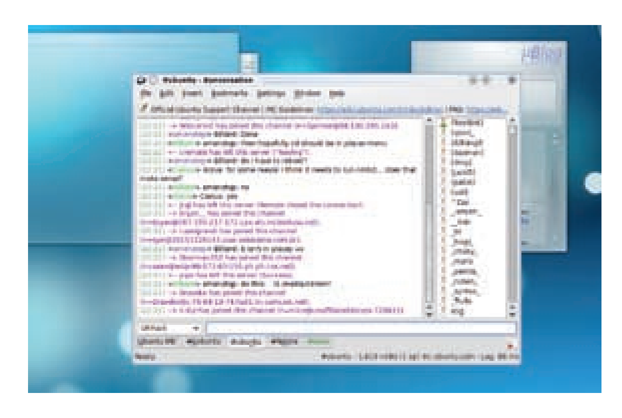
Konversation, естественно, требует библиотек KDE; однако предназначен отнюдь не только для KDE. В Gnome он тоже весьма неплохо смотрится: это одно из немногих приложений, отвергающих дискриминацию по рабочему столу. Вдобавок надо будет обзавестись *CMake*, чтобы скомпилировать необходимые бинарники.

После этого *Konversation* будет готов к работе и появится в меню Интернет. При его первом запуске надо будет добавить базовые сведения об IRC-сети, поскольку *Konversation* является в виде чистого холста.

Выберите сервер

В первую очередь вам придется указать данные сервера, с которым вы намерены соединяться. Если вы интересуетесь Linux-тематикой, воспользуйтесь Freenode. Вам также будет предложено набрать свои сетевое имя [Nick] и пароль, чтобы не делать этого при каждом входе на Nickserv.

Огромное количество параметров настройки – в общем-то стереотип для KDE, но в столь простом приложении оно все равно потрясает. Особенно полезной показалась нам функция автозамены. Она основана на регулярных выражениях и позволяет *Konversation* конвертировать часто используемые строки в нечто более осмысленное. Конечно, чтобы использовать ее по максимуму, требуется хорошее знание регулярных выражений; если вы давно собирались познакомиться с ними, то *Konversation* – веская причина это сделать.



› *Konversation* **вносит элемент упорядоченности в мир IRC.**

> «Огромное число параметров настройки просто потрясает.»

Интерфейс облегчает задачу переключения между каналами и серверами, а ведение журнала. *Konversation* способствует укреплению вашей организованности; и мы обнаружили, что в работе он стабилен. Так что если вы хотите сменить IRC-клиент, рассмотрите кандидатуру *Konversation*.

Персональный финансовый менеджер

Opale



Версия 1.0 **Сайт** tinyurl.com/yza843x

В Linux хватает солидных персональных финансовых менеджеров, вроде пресловутого *GnuCash*, и они отлично справляются со своей работой. Однако иногда бывает надо просто глянуть на свои расходы, чтобы быстро оценить, куда ушли ваши потом и кровью заработанные денежки.

Мы уже рассказывали об *Opale*, но это было аж в **LXF79**, а недавний выход версии 1.0 означает, что пора взглянуть на достижения разработчика. Простота интерфейса осталась неизменной; и в вашем распоряжении все те же основы введения в приложение.

Сперва вам надо определить учетную запись; для этого нажмите на кнопку Создать [New] на панели инструментов. Затем по мере необходимости вводите транзакции. Не ждите ничего сверхъестественного, поскольку *Opale* предназначен для хранения минимума данных, необходимых для отслеживания ваших доходов и расходов. Вы можете присвоить транзакции категорию, определяемую внутри приложения. После ввода требуемой информации *Opale* перенесет ее в реестр, и ваш баланс соответственно изменится.

Постоянные расходы

Весьма интересно, что *Opale* позволяет настраивать шаблоны расходов, и если вы знаете, что на пиво у вас еженедельно уходит около 50 фунтов, то можете настроить *Opale* на постоянное списание этой суммы. *Opale* умеет создавать простой график уровней ваших расходов по времени и выводить некую статистику, распределив ваши доходы и расходы по категориям.

> «Opale умеет создавать простой график ваших расходов по времени.»



› *Opale* **предлагает различные способы отслеживания ваших финансов.**

Компиляция *Opale* проста, но вам понадобятся библиотеки *Qt* и KDE (версии разработчика), а также *CMake*. Лучше всего программа смотрится в KDE, однако пользователи Gnome тоже вряд ли сочтут ее чужеродной.

Хотя это и не наш любимый язык, мы смогли поработать с XML-файлом, который *Opale* задействует для каждой транзакции, и предлагаем вам пользоваться им для добавления категорий и типов транзакций; кроме того, мы обнаружили, что это – единственная возможность добавить счет в фунтах стерлингов. Не паникуйте заранее: синтаксис файла не особо сложен.

Редактор гитарной табулатуры

TuxGuitar

Версия 1.2 **Сайт** www.tuxguitar.com.ar

Мы тут в Linux Format все талантливые музыканты, и иногда занимаемся композицией и аранжировкой. В прошлом мы обращались к *Frescobaldi* в поисках возможности соединять крупные фрагменты партитуры (и в данном номере рассматриваем *LilyPond* на стр. 42). Однако написание табулатуры для гитары – дело другое, поскольку система знаков отличается от обычной партитуры. Кроме того, создание музыки для гитары может оказаться непосильной задачей для тех, кто не читает нот, с ходу их оттолкнув.

На наше счастье, существует *TuxGuitar* – это приложение создавалось с целью превратить написание музыки для гитары в удовольствие. Вместо классической нотной записи вам выдается набор позиций на грифе. Для новичков есть возможность не ограничиваться введением музыки исключительно в виде табулатуры, поскольку можно также использовать внешнюю MIDI-клавиатуру для наигрыша мелодии, которую вы намерены транскрибировать.

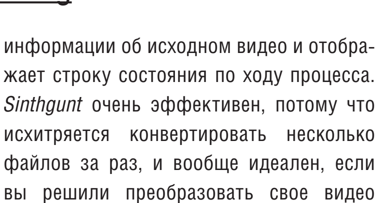
Тем, у кого, как и у нас, нет под рукой клавиатуры, предоставляется ее виртуальный аналог. Если он покажется чересчур хитроумным, можно использовать старую добрую нотную грамоту, которая сразу же превратится в табулатуру для гитаристов.

Для контроля за процессом, можно прослушивать свой шедевр по мере обретения им формы – и наблюдать за хаосом, что породили ваши пальцы, глядя на пляску нот вокруг грифа внизу экрана.



› Вводите музыку самыми разными способами и трансформируйте ее в табулатуру с помощью *TuxGuitar*.

Конвертор видео



Sinthgunt

Версия 2.0.2 **Сайт** www.sinthgunt.org

Да, мы за свободу ПО во всех ее проявлениях. Наличие достойных открытых стандартов позволяет легко переходить от одной программы к другой, но заметным исключением являются медиа-плейеры.

И хотя с аудиоформатом мы справляемся довольно легко, видео периодически причиняло нам головную боль – пока мы не наткнулись на *Sinthgunt*, графический интерфейс для *FFmpeg*. Он прост в использовании и требует только наличия *FFmpeg*, собираете ли вы его из исходников или берете один из готовых пакетов.

Вы можете взять локальный файл или загрузить видео FLV из сети и затем превратить его в дюжину различных форматов, удобно организованных для разных платформ, таких как BlackBerry, iPod, PS3 и т.д. Процесс прост: выберите файл, укажите стандарт, в который вы хотите конвертировать его, и нажмите на Go. Проще не бывает. *Sinthgunt* предоставляет уйму информации об исходном видео и отображает строку состояния по ходу процесса. *Sinthgunt* очень эффективен, потому что исхитряется конвертировать несколько файлов за раз, и вообще идеален, если вы решили преобразовать свое видео в более дружественный формат.

А уж совершенно замечательна возможность создавать предпросмотр, чтобы вы увидели конечный результат, который вас ждет. Кроме того, можно бросить последний взгляд на исходный файл, чтобы сравнить его качество с тем, что вы получаете на выходе. LXF



› *Sinthgunt* умеет конвертировать файлы в длинный список разных форматов.

Также вышли

Новые и обновленные программы, тоже достойные внимания...

Lorem Ipsum Generator 0.3

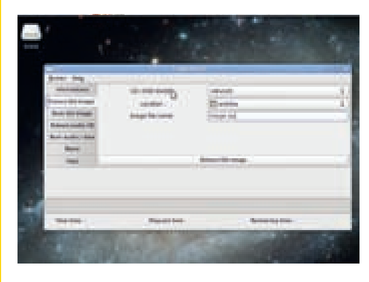
Вам нужен текст-заполнитель, причем в большом количестве? Используйте этот удобный инструмент, чтобы создавать строку за строкой полной белиберды.
http://code.google.com/p/loremipsum-generator



› *Lorem Ipsum Generator* создает поток текста, заполняющего пустое пространство.

SimpleBurn 1.2.3

Простая программа прожига дисков с возможностью создавать ISO-файлы и записывать их на носитель.
http://sallu.tuxfamily.org



› Используйте опрятный GUI SimpleBurn для всех задач, связанных с записью дисков.

Treeline 1.2.4

Полезное приложение, позволяющее навести порядок в ваших записях.
http://treeline.bellz.org

Gnumeric 1.8.4

Популярная альтернатива *OpenOffice.org Calc*, со множеством прелестей Gnome.
www.gnome.org/projects/gnumeric

Font Manager 0.4.1

Управляйте шрифтами с легкостью необычайной, при помощи этого удобного инструмента.
http://code.google.com/p/fontmanager

Drivel 3.0

Подчиняя себе Web 2.0, используйте Drivel, чтобы ваш блог был постоянно свежим и обновленным.
http://drivel.sourceforge.net

Два классных дистрибутива и гораздо больше отменных программ...



Читайте, чтобы найти всю информацию, которая вам понадобится для использования этого DVD! Если вы новичок в Linux, откройте index.html на диске (Сторона 1) и перейдите в раздел Справка: там вы найдете мини-учебник, дающий ответы на вопросы справа.

Майк Сондерс, редактор диска
mike.saunders@futurenet.com

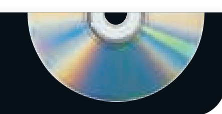
- » Что такое Linux?
- » Что такое дистрибутив?
- » Загрузка ПК с DVD
- » Разбиение жесткого диска на разделы
- » Навигация по файловой системе
- » Использование командной строки
- » Установка программ

Дистрибутивы Linux

Mandriva 2010 Free

Скажем сразу: это, возможно, самая простая из когда-либо созданных версий Linux. Если вы только знакомитесь с ОС и вас пугают разговоры о командной строке и компиляции программ, отбросьте свои страхи: Mandriva принесет на ваш компьютер всю мощь свободного ПО безо всяких проблем. На Стороне 2 **LXF**DVD находится полная, 4,3-ГБ версия Mandriva 2010 Free, укомплектованная настольными приложениями – инструментами для работы в Интернете, офисными программами, медиа-плейерами и многим другим – и все они готовы к работе.







Перед тем, как вставить DVD в дисковод, пожалуйста, убедитесь, что вы прочитали, поняли и согласились с нижеследующим:

Диски Linux Format DVD тщательно проверяются на предмет отсутствия на них всех известных вирусов. Тем не менее, мы рекомендуем вам всегда проверять любые новые программы надежным и современным антивирусом.

Хотя процесс отбора, тестирования и установки программ на DVD проводится со всем тщанием, редакция Linux Format не несет никакой ответственности за повреждение и/или утрату данных или системы, могущее произойти при использовании данного диска, программ или данных на нем. Настоятельно рекомендуем вам создавать своевременные и надежные резервные копии всех важных файлов.

Чтобы узнать об условиях использования, просим вас прочесть лицензии.

Бракованные диски

В маловероятном случае обнаружения бракованного диска Linux Format, просим связаться с нашей группой поддержки по адресу disks@linuxformat.ru для получения содействия.

По части оборудования, минимальные требования – 1-ГГц процессор, 512 МБ ОЗУ и 20 ГБ жесткого диска. Если вы работаете в Windows, то программа установки Mandriva позволит вам потеснить ее, чтобы выделить место для Linux, при условии, что в разделе Windows достаточно свободного пространства. Но обязательно создайте сначала резервные копии важных файлов Windows! После установки у вас появится программа-загрузчик, позволяющая выбирать между Windows или Linux при старте компьютера .

Для установки Mandriva следуйте приведенным ниже несложным инструкциям.

» Запустите компьютер со Стороны 2 DVD (при необходимости, измените порядок загрузки в настройках BIOS).

» Появившаяся программа установки (инсталлятор) предложит вам выбрать язык. Учтите: в нижней левой части экрана есть кнопка Справка [Help]. Она сопровождает практически каждый шаг процесса установки и выдает массу подробной информации; если это ваше первое свидание с Linux, прочтите подсказки полностью.

» Выберите раскладку клавиатуры, и вы перейдете к стадии разбиения диска на разделы. Здесь можно отвести весь диск целиком под Mandriva (это самый простой путь, если больше вам ничего на нем не нужно), изменить размер раздела Windows, чтобы высвободить место для Linux, или разбить диск на разделы вручную. Если вы изменяете размер раздела Windows, рекомендуем отвести под Linux не менее 20 ГБ, чтобы хватило на ОС и ваши личные файлы.

» На следующем шаге выберите Нет [None] на вопрос о дополнительных установочных носителях, затем назначьте рабочий стол (если вы сомневаетесь в выборе, рекомендуем KDE). После этого файлы Mandriva примутся копироваться на ваш жесткий диск, что может потребовать от 10 до 30 минут.

» Затем вам предложат задать пароль администратора (root) и создать учетную запись обычного пользователя. Помните, что в имени пользователя и пароле важен регистр.

» И, наконец, программа установки предложит записать загрузчик (наилучший вариант – выбор по умолчанию, Master Boot Record), затем выдаст вам обзор всех настроек на случай, если вы решите что-то изменить.

После этого инсталлятор перезагрузит ваш компьютер – извлеките DVD, и вы запустите Mandriva с жесткого диска. Наслаждайтесь знакомством с программами, а если понадобится помощь, загляните на www.mandriva.ru и www.linuxforum.ru.



› Программа установки Mandriva проста, а подробная подсказка вызывается одним щелчком.

Дистрибутив Linux

OpenSUSE 11.2

Еще один хит в нашем параде дистрибутивов – OpenSUSE 11.2. SUSE Linux существует с 1994 года и объединяет совершенную немецкую инженерию с мощным инструментом настройки, *Yast*, который стал самым известным компонентом дистрибутива. Поклонники OpenSUSE любят эту систему за внимание к деталям, а по мнению редакции **LXF**, OpenSUSE также – лучший выбор, если вы ищете себе дистрибутив с рабочим столом KDE.

На Стороне 1 **LXF**DVD этого месяца вы можете найти новый релиз OpenSUSE 11.2. Просто загрузите компьютер с диска в приводе и выберите OpenSUSE в загрузочном меню. Системные требования для хорошей производительности: 512 МБ ОЗУ, 3 ГБ места на жестком диске и 1-ГГц+ процессор.

Обратите внимание, что OpenSUSE загружается в режиме Live, и вы сможете познакомиться с рабочим столом и входящими в него программами, не касаясь своего жесткого диска. Если раньше вы никогда не пользовались KDE 4, нажмите на выгнутый значок справа вверху, чтобы добавить виджеты – то есть небольшие апплеты – на свой рабочий стол.

Чтобы найти программы, кликните по значку с зеленым гекконом внизу слева и перейдите в нужную позицию в меню. Для доступа к *Yast*, системе настройки «все-в-одном», щелкните по вкладке Компьютер [Computer], и затем – Yast. Решив установить OpenSUSE на жесткий диск, кликните по значку Установить [Install] на рабочем столе (или перейдите в Компьютер > Установить [Computer > Live Installer] в меню приложений) и следуйте инструкциям.



› OpenSUSE зеленый. «О, Эдмунд, неужели это правда? И я действительно держу в своих смертных руках частицу истинной Зелени?»

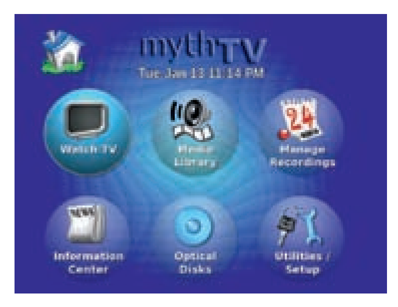
Если хотите узнать об OpenSUSE больше, загляните на сайт проекта на http://ru.opensuse.org — здесь найдутся подсказки, обновления и ссылки на другие ресурсы. В частности, форумы на http://forums.opensuse.org – отличное место, чтобы поделиться советом или получить помощь от других пользователей OpenSUSE. Как гласит классический слоган SUSE: Получайте удовольствие!

Другие программы

Tiny Core, MythTV, SeaMonkey

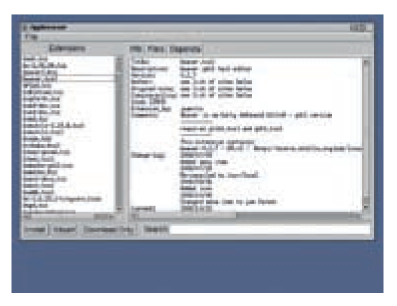
Стоит попробовайть еще один дистрибутив, в корне отличающийся от остальных: Tiny Core Linux – это 10-МБ микровариация Linux, загружающая чистый рабочий стол, а программы вы потом можете добавить через Интернет. Если вам нужно настроить защищенную Linux-машину, например, в качестве точки web-доступа в школе или публичной библиотеке, то это – отличный выбор: запишите и загрузите ISO-образ на **LXF**DVD, чтобы познакомиться с ним, или, если хотите сэкономить CD-R, испробуйте Tiny Core на виртуальной машине/эмуляторе ПК, например, *VirtualBox* или *Qemu*.

А еще в разделе **Рабочий стол** есть новый релиз MythTV, версию 0.22. Это приложения для записи цифрового видео было портировано на *Qt 4*, и теперь поддерживает видеоускорение VDPAU и работает с картами DVB-S2 и Hauppauge HD-PVR.



› Жаждете настроить личный цифровой видеомагнитофон? Наш ответ — MythTV!

Затем у нас имеется *SeaMonkey 2.0* (в разделе **Интернет**), крупное обновление пакета для работы во всемирной Сети от проекта Mozilla. Версия 2.0 соответствует *Firefox 3.5* по части кода и движка для рендеринга HTML, а вот среди новых функций – восстановление сессии (чтобы сохранить ваши окна и вкладки в неприкосновенности после сбоя), ведение журнала в базе данных и RSS-ленты в компоненте Mail/News. Распакуйте файл **seamonkey-2.0.1.tar.bz2** в домашнем каталоге и запустите **seamonkey** в получившейся директории.



› Tiny Core Linux вмещает самое насущное для дистрибутива Linux в 10 МБ, а остальное ищите в Интернете.

И это еще не все!
Подборка классных игр

В этом месяце в разделе **Игры** у нас гремучая смесь. Во-первых, здесь есть *Aviation*, новый имитатор полетов – он только-только достиг версии 0.1, но уже выглядит весьма впечатляюще. *Aviation* стремится к балансу между точным соответствием законам физики и притягательностью «простой игрушки», и смотрится просто отлично – 77-МБ файл **columbia-0.1.tar.bz2** содержит 600 квадратных миль местности вокруг реки Колумбия на северо-западе США.

Следом идет *LBreakout*, суперсовременное воплощение классического арканоида. Вы, возможно, считаете, что из такой идеи много не выжмешь, однако это не тот случай – в *LBreakout* есть растущие блоки, взрывающиеся блоки, регенерирующиеся блоки… И все это более чем на 50 уровнях, с большим количеством артефактов, увеличивающих и уменьшающих ваши возможности.

Затем есть *Xut*, любопытный маленький имитатор пуговичного футбола. Если вы сроду о таком не слышали (вне Бразилии в него не часто играют), можете представить его в виде игры в блошки, но только фигурки чуть более крупные и конусообразные. Так вот, *Xut* – это классно, и теперь он есть под Linux.



› Поднимитесь в небеса со своей благословенной стальной птицей в *Aviation*.

Приветствуем Karmic Koala, лучший из Ubuntu...

Дистрибутив Linux

Ubuntu 9.10

В сообществе Linux найдется немного новостей, которые порождали бы такой же ажиотаж, как новый релиз Ubuntu. 9.10, Karmic Koala, умудрился улучшить уже и так быструю загрузку 9.04, и дистрибутив заработал рейтинг 9/10 в нашем обзоре на стр. 8. Как и с предыдущими релизами, мы включили в DVD не просто обычный Ubuntu: мы отложили дистрибутив в сторонку, накопали огромное количество дополнительных программ и объединили все это вместе в специальную редакцию **LXF**. Вот что получите вы, и чего нет в обычном релизе:

» **Настольные приложения** *AbiWord*, *Gnumeric*, *Scribus*, *Inkscape*, *Blender*, *Gramps*, *GnuCash*, *HomeBank*.

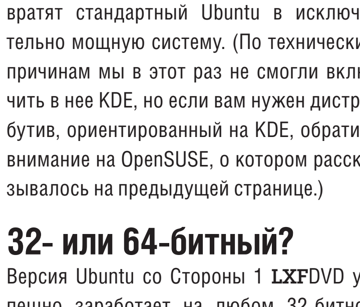
» **Программы для работы в Интернете** *Thunderbird*, *X-Chat*, *Epiphany*, *Pidgin*.

» **Игры** *Frozen Bubble*, *Wormux*, *SuperTux*, *NeverBall*, *Extreme Tux Racer*, *Freeciv*, *GCompris*.

» **Инструменты разработки** GCC, G++, *Gambas*, *Anjuta*, *MonoDevelop*, заголовки GTK/SDL и многое другое.

Всего здесь 300 пакетов, которые превратят стандартный Ubuntu в исключительно мощную систему. (По техническим причинам мы в этот раз не смогли включить в нее KDE, но если вам нужен дистрибутив, ориентированный на KDE, обратите внимание на OpenSUSE, о котором рассказывалось на предыдущей странице.)

› **Специальная LXF версия Ubuntu богата дополнительными программами: здесь есть *AbiWord*, *GnuCash* и *HomeBank*.**





› **Утилиту «Установка и удаление программ» заменил Центр приложений Ubuntu.**

32- или 64-битный?

Версия Ubuntu со Стороны 1 **LXF**DVD успешно заработает на любом 32-битном или 64-битном Intel-совместимом компьютере с такими системными требованиями:

» 512 МБ ОЗУ

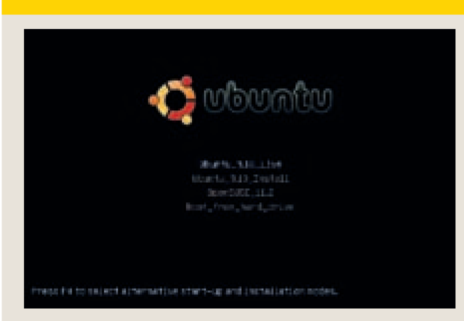
» 8 ГБ места на диске

» 1-ГГц+ процессор

На той стадии установки, когда винчестер разбивается на разделы, вы можете либо отвести под Ubuntu весь диск (самый простой выбор), либо разметить разделы вручную – например, если хотите, чтобы Ubuntu сосуществовал с другим дистрибутивом. Если вы используете Windows, программа установки позволит вам изменить размер раздела Windows, выделив место под Linux. Внимание: являясь в принципе безопасной операцией, разбиение диска на разделы все же может привести к потере данных, если что-то пойдет не так, поэтому первым делом сохраните все ваши важные файлы Windows! Следуйте инструкциям ниже, чтобы установить Ubuntu, и загляните на следующую страницу: там вы найдете подсказки и советы.

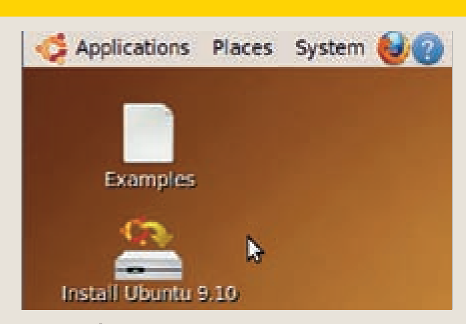
Если у вас машина x86-64 и вы хотите выжать из своего процессора всю возможную производительность до капли, запишите ISO-образ **amd64**, который также найдется на **LXF**DVD, и загрузитесь с него. Далее, если вы хотите настроить Ubuntu в качестве сервера – то есть вас не интересует графический рабочий стол – можете использовать ISO под названием **server**. Более подробная информация об этих версиях – на www.ubuntu.com.

Шаг за шагом: Устанавливаем Ubuntu 9.10



1 Загрузка

Запустите компьютер со Стороны 1 DVD и нажмите Enter, увидев это меню. (Возможно, сначала вам потребуется изменить порядок загрузки в BIOS, установив первым DVD вместо жесткого диска).



2 Рабочий стол

На рабочем столе можно поэкспериментировать с имеющимися программами из меню Приложения [Applications], или дважды щелкнуть по значку Install [Установить] для запуска программы установки.



3 Разбиение на разделы

Следуйте указаниям по настройке местоположения и раскладки клавиатуры. На экране с дисковым пространством можно изменить размер раздела Windows, чтобы выделить место для Linux, использовать весь диск целиком, или разметить разделы вручную.

Как бы мне?..

Если вы – полный чайник в мире Linux, после установки Ubuntu вы можете испытывать некоторую неуверенность в решении ряда задач. Вот список наиболее общих действий и способов их выполнения…

» **Работа в сети** Щелкните по значку с оранжево-голубым земным шаром на верхней панели – запустится *Firefox*, второй по популярности браузер в мире, поддерживающий сотни расширений.

» **Редактирование документов** Перейдите в меню Приложения > Офис [Applications > Office]; там вы найдете *OpenOffice.org*, мощный пакет, совместимый с документами *MS Office*. Он довольно сильно нагружает память, поэтому можете использовать его более легкие эквиваленты, *AbiWord* и *Gnumeric*.

» **Воспроизведение музыки и видео** В подменю Звук и видео [Sound & Video] вы найдете видеоплейер *Movie Player* и музыкальный проигрыватель *Rhythmbox*.

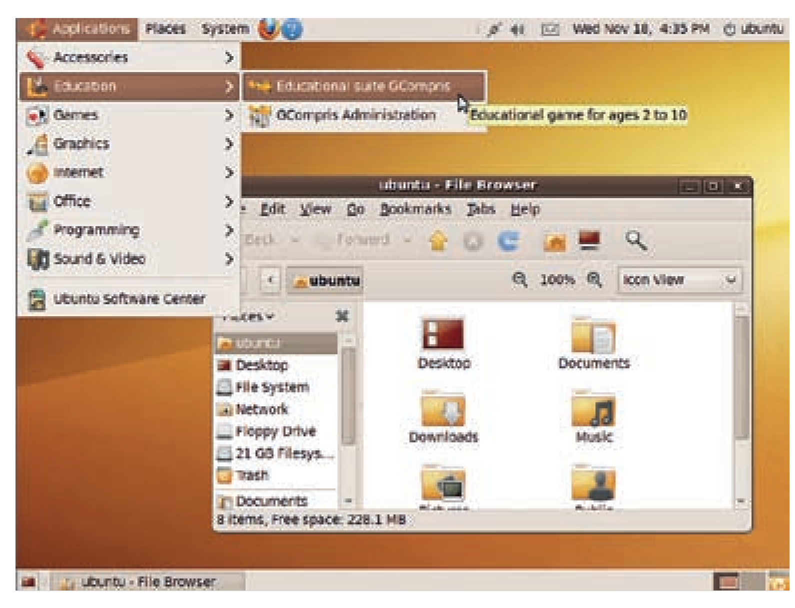
» **Чат онлайн** В Приложения > Интернет имеются *Empathy* и *Pidgin*, два интернет-пейджера, снабженных множеством функций. Попробуйте оба.

» **Настройка системы** В меню Система > Параметры [System > Preferences] полно опций рабочего стола, клавиатуры, энергосбережения и сети, а через Система > Администрирование [System > Administration] осуществляется управление пользователями, настройка оборудования и получение обновлений.

» **Поиск новых программ** Щелкните по Приложения > Центр приложений Ubuntu, чтобы загрузить программы из сети. (Более продвинутый инструмент – Система > Администрирование > Synaptic.)

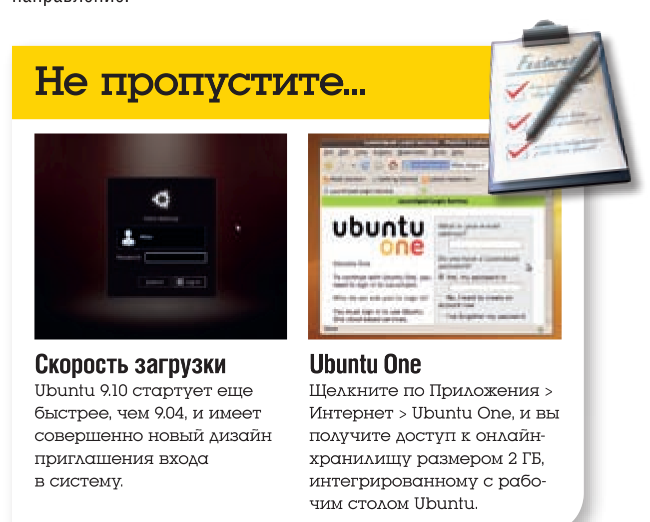
» **Выключение** Щелкните по значку включения, который располагается на верхней панели крайним справа.

Если вам нужна помощь по Ubuntu, щелкните по Система > Справка и поддержка [System > Help and Support] – там вы найдете указания по решению наиболее общих проблем, типа выхода в Интернет, импортирования фотоснимков и настройки принтеров/сканеров. Руководства Ubuntu написаны с учетом всех уровней пользователей, так что технические термины не поставят вас в тупик. Если ответа все же не нашлось, загляните на www.linuxforum.ru и www.ubuntu.ru. На этих форумах вы сможете разместить свое сообщение и получить помощь от других пользователей Ubuntu. Ясно изложите свою проблему и укажите все подробности сообщений об ошибке и о том, что вы пытались сделать, и кто-нибудь обязательно укажет вам верное направление.



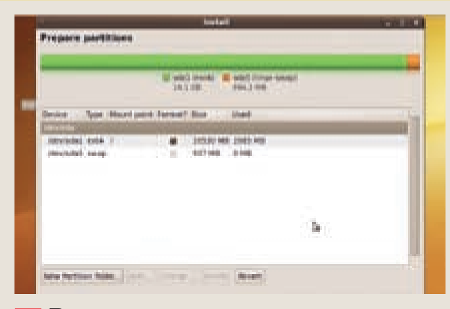
› **В том, что касается тем, Ubuntu 9.10 (Karmic Koala) возвращается к темно-коричневым тонам панелей заголовков, как в ранних релизах.**

Не пропустите...

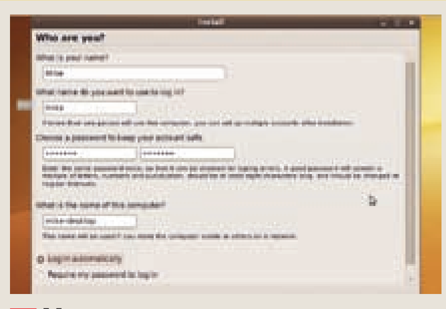


Скорость загрузки
Ubuntu 9.10 стартует еще быстрее, чем 9.04, и имеет совершенно новый дизайн приглашения входа в систему.

Ubuntu One
Щелкните по Приложения > Интернет > Ubuntu One, и вы получите доступ к онлайн-хранилищу размером 2 ГБ, интегрированному с рабочим столом Ubuntu.



4 Вручную
Если вы выделяете разделы вручную, рекомендуем создать корневой раздел в формате ext4 размером не менее 10 ГБ и раздел подкачки размером 1 ГБ.



5 Учетная запись
Затем задайте имя пользователя и пароль, идентифицирующие вас в системе, и укажите, хотите ли вы входить автоматически (В имени пользователя и пароле важен регистр!)



6 Подтверждение
Программа установки уточнит, хотите ли вы продвигаться дальше (щелкните по Дополнительно [Advanced], если вам нужно изменить параметры загрузчика). Затем файлы скопируются на жесткий диск, и система перезапустится. LXF

Информация о диске

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Внимательно прочтите это перед тем, как использовать LXFDVD!

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

» **имя_программы-1.0.1.i386.rpm** – вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;

» **имя_программы-1.0.1.i386.deb** – такой же пакет, но уже для Debian;

» **имя_программы-1.0.1.tar.gz** – обычно это исходный код;

» **имя_программы-1.0.1.tgz** – тот же файл, что и выше по списку: «tgz» – это сокращение от «tar.gz»;

» **имя_программы-1.0.1.tar.bz2** – тот же файл, но сжатый bzip2 вместо обычного gzip;

» **имя_программы-1.0.1.src.rpm** – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

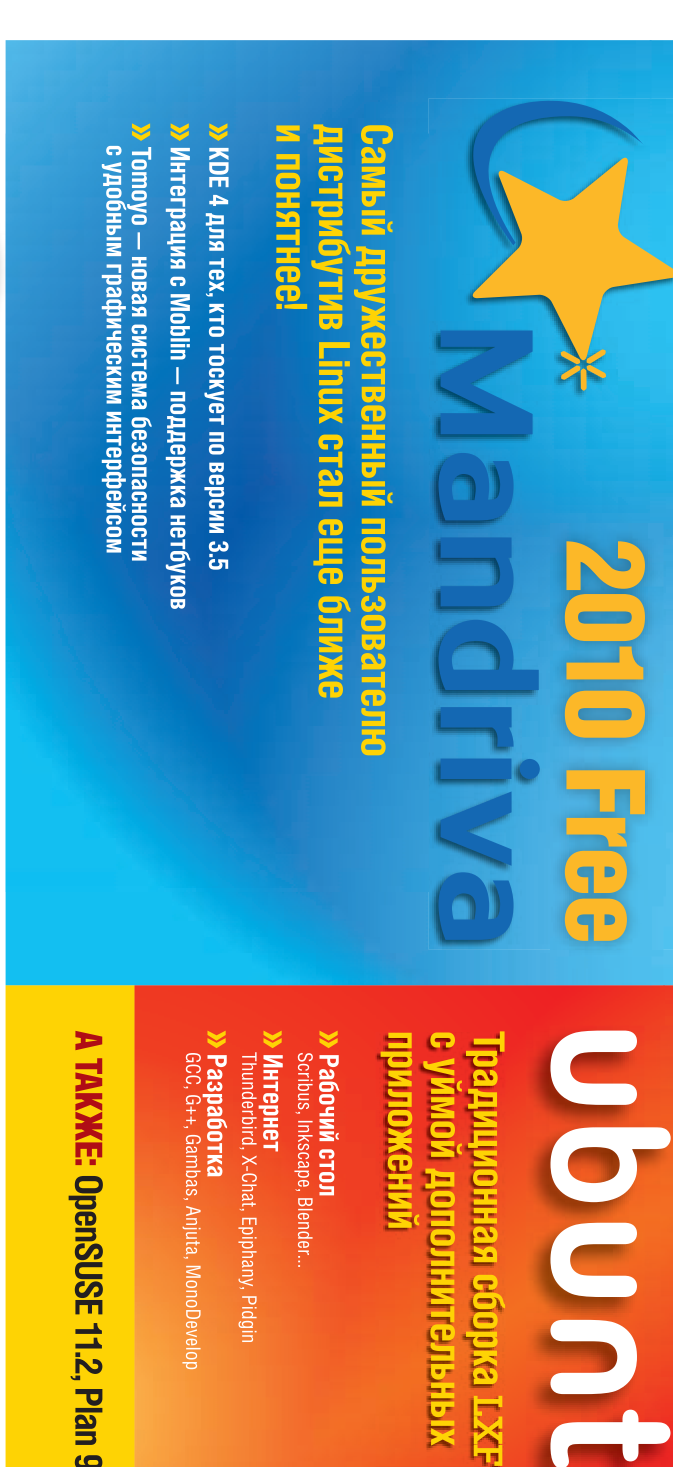
» **имя_программы-1.0.1.i386.FC4.RPM** – двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

» **имя_программы-1.0.1.ppc.Suse9.rpm** – двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

» **имя_программы-devel-1.0.1.i386.rpm** – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru



Январь 2010
LXFDVD 126/127

Январь 2010
LXFDVD 126/127



Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что если у вас нет устройства, с помощью которого можно было бы записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков: подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика.



DVD-приложение к журналу Linux Format № 1 (126/127). Январь 2010

Содержание

LINUX FORMAT

Сторона 1

Сторона 2

Пожалуйста, перед использованием данного диска ознакомьтесь с опубликованной в журнале инструкцией!

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь пожалуйста по адресу disks@linuxformat.ru

Тираж изготовлен ООО «Уральский электронный завод». 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПТР ВАФ 77-15.

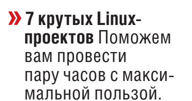


Пропустили номер?

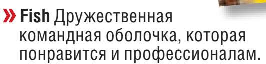
» Мир свободного ПО богат и разнообразен, а потому далеко не все можно вместить в рамки одной статьи. Linux Format обходит эту проблему, публикуя серии статей по самым актуальным вопросам, но что делать, если вы поймали интересующий вас материал на середине? Обратитесь в Линукс-центр по адресу www.linuxcenter.ru и закажите желаемый номер журнала! Он доставляется как в печатной, так и в электронной форме, поэтому с момента открытия браузера и до получения нужного вам выпуска **LXF** может пройти не более нескольких минут!

Прямо сейчас для заказа доступны следующие номера:

LXF123
Октябрь 2009



- **7 крутых Linux-проектов** Поможем вам провести пару часов с максимальной пользой.
- **Fish** Дружественная командная оболочка, которая понравится и профессионалам.
- **Blender** Нетривиальные техники трехмерного моделирования в свободном 3D-пакете.
- **Ekiga** Достойная альтернатива бесплатному Skype плюс видеоконференции.

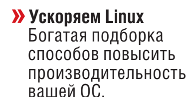


LXFDVD: CrunchBang Linux 9.04, MOPS Linux 6.2.2 и коллекция свободного ПО для Windows

Печатная версия:
http://www.linuxcenter.ru/shop/books-and-magazines/Linux-Format/lxf_123/

Электронная версия в формате PDF:
http://www.linuxcenter.ru/shop/electr/magazine/elxf_123/

LXF124
Ноябрь 2009



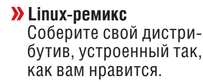
- **Ускоряем Linux** Богатая подборка способов повысить производительность вашей ОС.
- **BSD: приручаем демона** Введение во FreeBSD для завзятых линуксоидов.
- **Drupal: сайт легко и быстро** Познакомьтесь с самой популярной открытой CMS.
- **Kdenlive** Уроки видеомонтажа свободными инструментами в свободной среде.

LXFDVD: Быстрые дистрибутивы Linux, FreeBSD 7.2 и Calculate Linux 9.9

Печатная версия:
http://www.linuxcenter.ru/shop/books-and-magazines/Linux-Format/lxf_124/

Электронная версия в формате PDF:
http://www.linuxcenter.ru/shop/electr/magazine/elxf_124/

LXF125
Декабрь 2009



- **Linux-ремикс** Соберите свой дистрибутив, устроенный так, как вам нравится.
- **Виртуализация — это просто** KVM, libvirt и другие современные технологии.
- **TaskJuggler** История одного проекта и вводный курс по управлению ими.
- **Yum и PackageKit** Уроки пакетного менеджмента в дистрибутивах Red Hat и не только.

LXFDVD: Slackware 13.0, Zenwalk 6.2 и ALT Linux 5.0

Печатная версия:
http://www.linuxcenter.ru/shop/books-and-magazines/Linux-Format/lxf_125/

Электронная версия в формате PDF:
http://www.linuxcenter.ru/shop/electr/magazine/elxf_125/

Ну, а если вы хотите быть уверенными, что не пропустите ни один номер журнала – оформите подписку! Помните, что все подписавшиеся на печатную версию журнала через www.linuxcenter.ru получают электронную версию в подарок!

Спешите на www.linuxformat.ru/subscribe!

Исполнители и алгоритмы

Говорят, в английском языке все слова взяты из C++. **Александр Казанцев** идет в обратном направлении и разбирается, как можно применить в программировании язык русский.



Наш эксперт

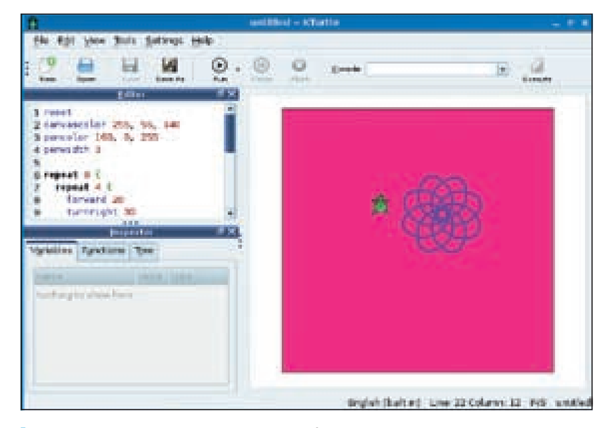
Александр Казанцев
К.т.н., доцент кафедры Информатики Глазовского государственного пединститута, руководитель проекта EduMandriva, автор локализаций (и разработок) для Mandriva, *Klavaro*, *Gambas2* и *LXDE*.

Информатика — наука сложная. С этим трудно поспорить. Особенно нелегко тем людям, у которых есть проблемы с логическим построением каких-либо задач. Логика лежит в основе множества составных частей информатики, а особенно — программирования. Последнее непосредственно связано с прикладной частью информатики. Получается замкнутый круг, из которого нужно искать выход.

С другой стороны, само программирование таит в себе скрытые проблемы – информатика из академической науки, сродни математике, превращается во всеобъемлющий инструмент для применения в других областях – и вот мы видим, что она востребована во многих возрастных группах и предметах: на русском языке, географии, физике, рисовании, музыке, в детском саду и начальной школе.

Вы можете возразить: а как это связано с проблемами программирования? Да очень просто. Если вернуться к определениям, то программа – это алгоритм, записанный на каком-либо языке. Этот язык по большей степени непонятен целевым группам (ради интереса подсуньте программу на C++ учителю музыки или четырехлетнему ребенку). С другой стороны, алгоритм как порядок действий является достаточно понятной вещью, и человек может составить его после непродолжительного этапа даже не обучения, а пояснения порядка действий (опять же получается каламбур: порядок действий для составления порядка действий). Но здесь мы упираемся в проблему, упомянутую на первом этапе – логику. Теперь вам все понятно?

Судя по вашим округлившимся глазам, уследить за полетом моей мысли смогли не все, поэтому не буду загружать вас «академическими россказнями» (поверьте, это я умею), а перейду непосредственно к делу.



› Красивый цветочек — но если забыть пакет локализации KDE, то черепашка будет понимать только английский язык.

Исполнитель и псевдокод = ?

Обратившись к Википедии, вы можете легко найти определения и первого, и второго. Из них можно заключить, что псевдокод – компактный способ описания алгоритмов, использующий ключевые слова языков программирования, но опускающий несущественные подробности и специфический синтаксис. Псевдокод содержит основные конструкции языков программирования и иногда даже повторяет их – тогда говорят о Паскаль- или Бейсик-подобном псевдокоде и т. п. – или пишется на естественном языке. Одним из первых языков программирования и псевдокодом одновременно был Алгол (сокращение от английских algorithmic – алгоритмический и language – язык) – дедушка всех императивных языков. Самым известным псевдокодом в России является разработанный академиком А.П. Ершовым в середине 1980-х годов школьный алгоритмический язык, который изначально был призван решить проблему безмашинного обучения, но после был применен А.Г. Кушниренко для управления исполнителями.

Исполнители

Итак, что поможет нам развить логическое мышление, познакомить с понятием алгоритма и даже обучить программированию? Ответ достаточно прозаичен и известен многим – это исполнители (роботы) и псевдокод. Первые выполняют роль посредников между пониманием команд, отдаваемых машине, и получаемым

результатом; второй помогает постичь саму суть алгоритма. Так как и первые, и вторые появились не прямо сейчас и не сегодня, то в настоящий момент существуют и исполнители, и псевдокоды. И их использование при обучении не только рекомендуется, но и требуется (например, не забегая вперед, скажем что один из вариантов псевдокода входит в задания Единого государственного экзамена по информатике). Опять же, исполнители не всегда будут работать под управлением псевдокода.

И вновь (наверное, я уже замучил вас затянувшейся интригой и постоянным подыманием проблем?): что делать при использовании свободного ПО и Linux в частности? Что может предложить нам мир свободы и доступности кодов? Давайте рассмотрим все по порядку.

Исполнители имеют одно общее свойство – у них есть определенный набор команд, который они понимают. Они бывают двух видов: исполнители «обстановки» и исполнители «величин». Давайте рассмотрим, какие исполнители мы можем найти «на просторах» открытого и свободного ПО.

Черепашка и все, все, все

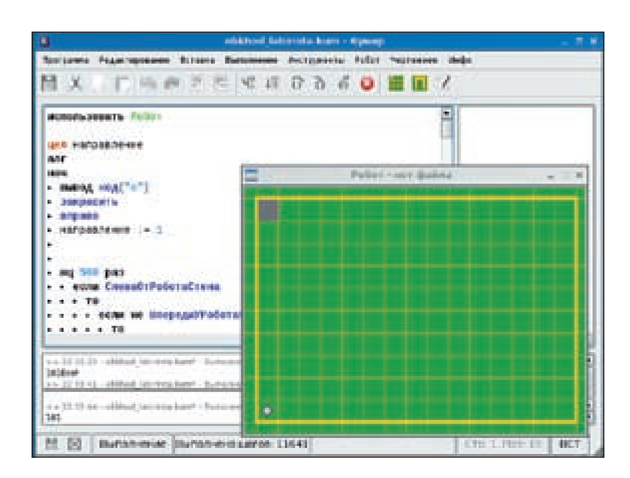
Самым известным исполнителем является «черепашка» (turtle). Она появилась в результате разработки педагогического программного средства обучения детей алгоритмизации психолога Сеймура Пейперта [Seymour Papert] – языка программирования Лого. Иными словами, черепашка является исполнителем алгоритмов, записанных на языке Лого.

В Linux также можно найти «черепашку» – приложение *KTurtle* является частью пакета образовательных приложений *KDE-Edu* и существует как для третьей, так и для новой, четвертой версии этого рабочего стола. Даже если у вас установлен дистрибутив, который основан не на KDE (например, Ubuntu), вы можете подключить репозиторий, содержащий KDE, и доустановить пакеты. Не забудьте также добавить пакеты локализации для KDE, иначе придется изучать язык оригинала.

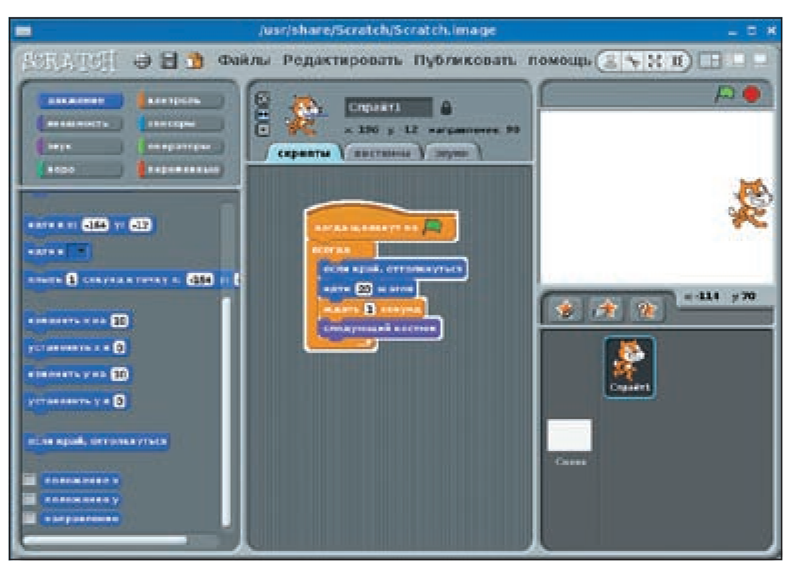
> «Для продвинутых существует библиотека черепашьей графики.»

Для более продвинутых пользователей (а также тех, у кого проблемы с приложениями KDE или кто не хочет тянуть «лишние» библиотеки) существует библиотека «черепашьей графики» для Python с понятным названием *turtle* (http://docs.python.org/library/turtle.html).

Черепашка оказала большое влияние на развитие другого проекта – *КуМир*. Исполнители Робот и Чертежник – по сути, Черепашка, разделенная на две составляющие. Хотя *КуМир* и был создан во второй половине 80-х годов, в настоящий момент идет активная работа над его продолжателем, который использует библиотеку *Qt* и поддерживает операционные системы GNU/Linux и Windows. Разработка ведется в НИИСИ РАН под руководством М. А. Ройтберга. С сайта проекта (http://lpm.org.ru/kumir/) вы можете скачать последнюю версию пакетов (а также версию для разработчиков, которая уже содержит более двух исполнителей) под известные дистрибутивы Linux, или исходные коды для самостоятельной сборки.



› Робот будет долго искать выход из этого «лабиринта».

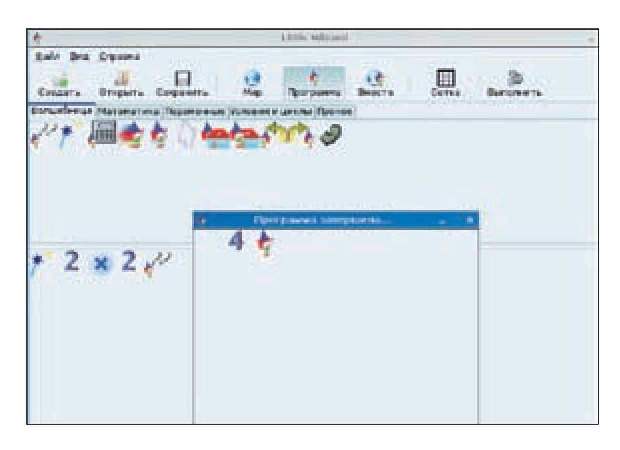


› Даже такая маленькая программа может заставить кота шагать от стенки к стенке.

В отличие от «черепашки», исполнители *КуМир*'а используют алгоритмический язык, который является частью заданий Единого государственного экзамена, что позволяет безболезненно и углубленно вписать их в программу обучения.

Еще более продвинутыми являются исполнители, программный код к которым написан на другом учебном диалекте «большого» языка программирования, SmallTalk. Мы говорим об учебных проектах, которые основаны на языке *Squeak*. Пожалуй, самым известными исполнителем, который написан на этом языке, является *Scratch* (http://scratch.mit.edu) – проект создания программ на диалекте языка Лого (то есть можно применять навыки, полученные при работе с Черепашкой), но без использования непосредственной записи алгоритмов. В *Scratch* программа набирается из «кирпичиков», как в конструкторе (авторы проекта не скрывают, что черпали идеи в конструкторе Лего и специализированном синтаксисе Лего-Лого), где строительными блоками являются команды действий исполнителя, работа с переменными, отслеживание событий (как во взрослых объектных языках программирования), циклы, условия, переходы и т. д. В *Scratch* можно создавать фильмы, играть с различными объектами, видоизменять их, перемещать по экрану, устанавливать формы взаимодействия между объектами. То есть можно сказать, что это объектно-ориентированная среда, в которой блоки программ собираются из разноцветных кирпичиков-команд. Поэтому вы можете использовать *Scratch* как в пропедевтических курсах в младших классах, так и на уроках программирования в старших – возрастная категория достаточно обширна. *Scratch* также будет хорошей заменой роликам на основе проприетарных Flash и Action Script – создавать мультфильмы, размещать их на сайте проекта в Интернете и показывать родителям, знакомым и друзьям здесь можно за пару щелчков мыши.

»



› 2×2 = 4, не так ли?

Вы можете найти последнюю версию *Scratch* в учебных редакциях ваших любимых дистрибутивов, или скачать пакеты (к сожалению, только в формате .deb) с сайта проекта http://info.scratch.mit.edu/Linux_installer. Там же можно прочитать, как установить *Scratch* из исходных текстов с помощью специального скрипта на Perl.

Что есть еще?

Можно упомянуть и другие исполнители, которые не столь известны, но заслуживают вашего внимания. В первую очередь это Маленькая Волшебница – *Little Wizard*. Она позволяет создавать программы для исполнителя с помощью кубиков-команд с рисунками. То есть воспользоваться этой программой могут самые маленькие «программисты», которые только учатся читать и писать. С помощью Миров можно построить новый дом, провести волшебницу по созданному лабиринту или дорожке. Причем алгоритмы могут быть достаточно сложными: есть даже циклы, позволяющие избежать дублирования похожих участков кода. Два режима – Числа и Алфавит – помогут составлять примеры и выводить текст. А поддержка ветвлений еще больше усложнит программный продукт.

> «Позволяет создавать программы с помощью кубиков-команд.»

› Битва в самом разгаре. Кто победит?

Little Wizard можно найти в репозиториях вашего дистрибутива или загрузить на сайте проекта – http://littlewizard.sourceforge.net/.

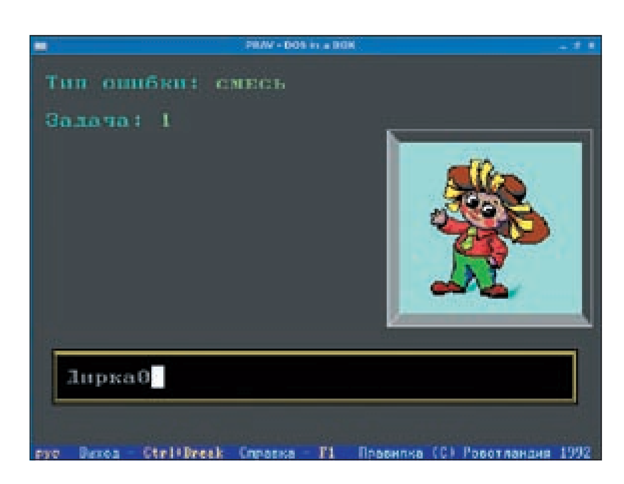
Кто-то скажет, мол, все это – для малышни, а есть ли в Linux что-то более сложное? Если коротко, то да. Одним из примеров «сложного» исполнителя является *Robocode* (http://robocode.sourceforge.net/). Сказать по правде, это даже не исполнитель, а поле боя для программистов. Каждый его участник пишет на Java код, управляющий небольшим танком-роботом, который сражается с другими такими же танками. Основная цель – уничтожить все остальные танки на поле. Без компромиссов. А при чем тут обучение, и как согласуются война и школа? Ничего страшного – в *Robocode* нет крови, нет людей и нет политики. В игре есть взрывы; но если вы заботитесь о психическом здоровье школьников, их можно легко отключить. У танков большие возможности: они могут двигаться по игровому полю, стрелять, узнавать местоположение друг друга, врезаться в стены и в других роботов, выяснять, где находятся летящие снаряды, и использовать всю мощь языка Java. Естественно, какой-либо простой стратегии для выигрыша нет. Существует огромное количество различных тактик, у каждой из которых есть свои преимущества и недостатки.

А как быть с проверенным ПО?

Уже слышу хор возмущений: «Подсунули тут неизвестные продукты!» Да, в настоящий момент, не считая *КуМир*'а, под Linux не существует родных версий столь известных вещей, как *Роботландия*, *ЛогоМиры* или *Паркетчик*. Что делать, если у вас есть копии этих продуктов, причем лицензионные? Сразу предупредим: все описанные далее программы имеют или закрытый код, или проприетарную лицензию, но опять же аналогов им на настоящий момент не существует (надеемся, эта статья послужит для кого нибудь посылом к действию).

Ничего страшного. Начнем с самого известного (к сожалению, в настоящий момент – во все более узких кругах) пакета программных продуктов «Роботландия». Он разработан еще под DOS, поэтому его выполнение в среде *Wine* невозможно. Но это не означает, что с ним нельзя работать в Linux (на вопрос о том, надо ли использовать такой старый продукт, я обычно отсылаю узнать возраст *Vi* и *Bash*).

Запуск *Роботландии* отлично реализуется с помощью DOS-эмулятора *dosemu* и установленной в нем свободной версии DOS – FreeDOS. Вы также можете найти их в репозиториях вашего дистрибутива. Установив эмулятор, вы получите прекрасно работающую DOS-среду, в которой можно запустить *Роботландию* (а также старого доброго *КуМир'а*, кросплатформенная версия ко-



› Знакомая «Правилка», но под Linux.

торого не доросла до уровня DOS-редакции, *Кенгуренка*, и другие хорошие проверенные обучающие продукты, которые, к сожалению, почили в веках). Затем вы сможете проверить работу исполнителей *Роботландии*, и наконец-то познакомиться с предметом обожания всех маленьких компьютерщиков начала 90-х – Петей Куком и его друзьями (и не только с исполнителями), проживающими на сайте http://www.botik.ru/~robot.

Там же вы найдете «роботландский» аналог «черепашки» – *Кукарачу*, которая идет в среде *Wine* и (к сожалению) не имеет ничего общего со свободным ПО.

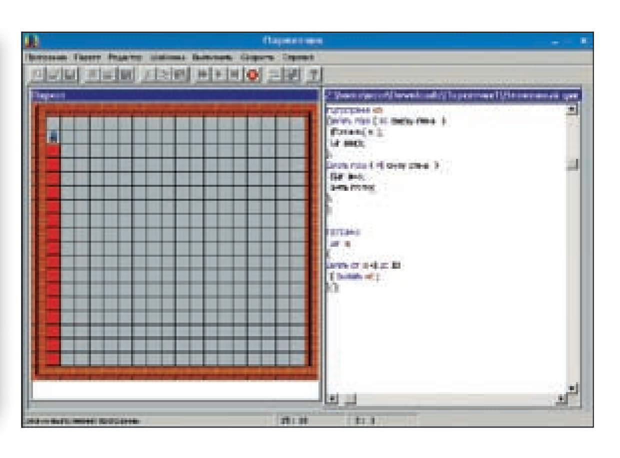
Паркетчик (его вы встретите в учебниках информатики за авторством А. Г. Гейна) представляет собой исполнителя, который, в отличие от «черепашки», не рисует, а выкладывает плитку (нет, ремонт в ванной и на кухне он сделать не поможет). Управление исполнителем ведется на основе Паскаль-подобного псевдокода на русском языке. Этот исполнитель также прекрасно работает в среде *Wine*, и можно скачать его бесплатно распространяемую версию с http://www.prosv.ru/info.aspx?ob_no=19088, где есть и более сложные исполнители – машины Поста и Тьюринга.

ЛогоМиры также прекрасно чувствуют себя в *Wine*. И даже более того – разработчики обещают скоро выпустить новую версию продукта (под кодовым именем «Пломбир»), которая будет считать Linux своим родным домом. Поэтому повсеместное внедрение и использование Linux будет как раз тем стимулом, который подстегнет разработчиков создавать, портировать и адаптировать свои продукты под Linux, а возможно, вступать в дружный круг создателей свободного ПО, так как идеи, витающие в нем, как раз совпадают с идеями образования. LXF

Полезные ссылки

Чтобы не заставлять вас рыскать по просторам Интернета, мы сделали подборку сведений по исполнителям, которая, надеемся, будет актуальна в ближайшее время.

- Руководство по *KTurtle* (если вы не поставите себе KDE) – http://docs.kde.org/stable/ru/kdeedu/kturtle/index.html
- Алгоритмизация и основы программирования на базе *KTurtle*: Учебное пособие – http://window.edu.ru/window/catalog?p_rid=58396
- *Scratch* на Летописях.ру – http://letopisi.ru/index.php/Скретч
- Учитесь со *Scratch* – http://setilab.ru/scratch/
- Программируем на Лого и Python – http://freecode.pspo.perm.ru/115/
- *LittleWizard* – http://freecode.pspo.perm.ru/236/littleWizard/index.html



«Ложит» плитку, а потом перекладывает снова. Ничего не напоминает?..



Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 5000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор
Валентин Синицын info@linuxformat.ru
Литературный редактор
Елена Толстякова
Переводчики
Илья Аввакумов, Александр Бикмеев, Юлия Дронова, Светлана Кривошеина, Александр Казанцев, Алексей Опарин, Валентин Развозжаев, Татьяна Цыганова
Редактор диска
Александр Кузьменков
Верстка, допечатная подготовка
Сергей Рогожников
Технический директор
Денис Филиппов
Генеральный директор
Павел Фролов
Учредители
Частные лица
Издатели
Виктор Федосеев, Павел Фролов

Отпечатано в типографии «Взлет»
196644, Санкт-Петербург, Колпинский район, пос. Саперный, тел. (812) 462-85-85
Заказ

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Пол Хадсон [Paul Hudson] paul.hudson@futurenet.com
Редактор обзоров Грэм Моррисон [Graham Morrison] graham.morrison@futurenet.com
Редактор диска Майк Сондерс [Mike Saunders] mike.saunders@futurenet.com
Художественный редактор Эфраин Эрнандес-Мендоса [Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com
Литературный редактор Эндрю Грегори [Andrew Gregory] agregory@futurenet.com
Штатный автор Боб Мосс [Bob Moss] bob.moss@futurenet.com
Подготовка материалов
Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Энди Ченнел [Andy Channelle], Карл Флетчер [Carl Fletcher], Майкл Дж. Хэммел [Michael J. Hammel], Энди Хадсон [Andy Hudson], Джон Хадсон [John Hudson], Мартин Мередит [Martin Meredith], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Ник Вейч [Nick Veitch], Евгений Балдин, Антон Борисов, Андрей Боровский, Семен Есилевский, Александр Казанцев, Евгений Крестников, Дмитрий Попов, Александр Толстой, Алексей Федорчук, Роман Ярыженко
Художественные ассистенты: Ник Эспелл [Nick Aspell], Фил Хейкрэфт [Phil Haycraft], Сэлли Хендерсон [Sally Henderson], Сара Николь [Sarah Nicol]
Иллюстрации: Крис Хедли [Chris Hedley], Крис Винн [Chris Winn]

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel +44 01225 442244 Email: linuxformat@futurenet.com
РОССИЯ:
Санкт-Петербург (редакция):
Лиговский пр., 50, корп. 15
Тел.: +7 (812) 309-06-86
Представительство в Москве:
Красноказарменная ул., 17, м. «Авиамоторная» (в помещении АТС МЭИ)
Тел/факс: +7 (499) 271-49-54
По вопросам сотрудничества, партнерства, оптовых закупок:
partner@linuxcenter.ru



BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO



В февральском номере

Начните работать в Linux

Кто-то из ваших знакомых или друзей интересуется свободным ПО, но боится попробовать? Не беда – просто дайте ему или ей почитать этот номер.

Восстановим разделы

Люди делятся на две категории: тех, кто не делает резервные копии, и тех, кто уже делает. Если вам «посчастливилось» попасть в первую, то узнать, как вернуть столь некстати пропавшие данные, будет явно не лишним.

Проекты на 90 минут

Мы неоднократно помогали вам провести время с пользой, и готовы делать это вновь и вновь. Разверните собственное web-приложение, попробуйте *Gnome Shell* или просто послушайте хорошую музыку.

Содержание последующих выпусков может меняться без уведомления.

wimaxstore РЕКОМЕНДУЕТ

Mobile WiMax

ИНТЕРНЕТ НОВЕЙШЕГО ПОКОЛЕНИЯ 4G

Интернет WiMax – это:

- Высокоскоростной доступ — до 10 Мбит/с в любой точке зоны покрытия
- Связь в тех местах, которые раньше были вне досягаемости широкополосного доступа
- Легкое и быстрое подключение
- Не требуется прокладки проводов и, соответственно, дополнительных инвестиций
- Triple play: возможность интеграции разнородного трафика (данные, голос, видео)
- Круглосуточная техническая поддержка без выходных





1 РОУТЕР **ASUS WL 500GP**
\+ МОДЕМ SAMSUNG SWC-U200
6940 руб.

2 РОУТЕР **D-LINK DIR-320**
\+ МОДЕМ SAMSUNG SWC-U200
5250 руб.



САНКТ-ПЕТЕРБУРГ **+7 812 309-06-86**

МОСКВА **+7 499 271-49-54**

WWW.WIMAXSTORE.RU

Технические специалисты WiMax Store разработают для вас заказные программно-аппаратные решения любой сложности на базе WiMax. Звоните сегодня!

Цены указаны с учетом НДС.